ENCICLOPEDIA PRACTICA DE LA

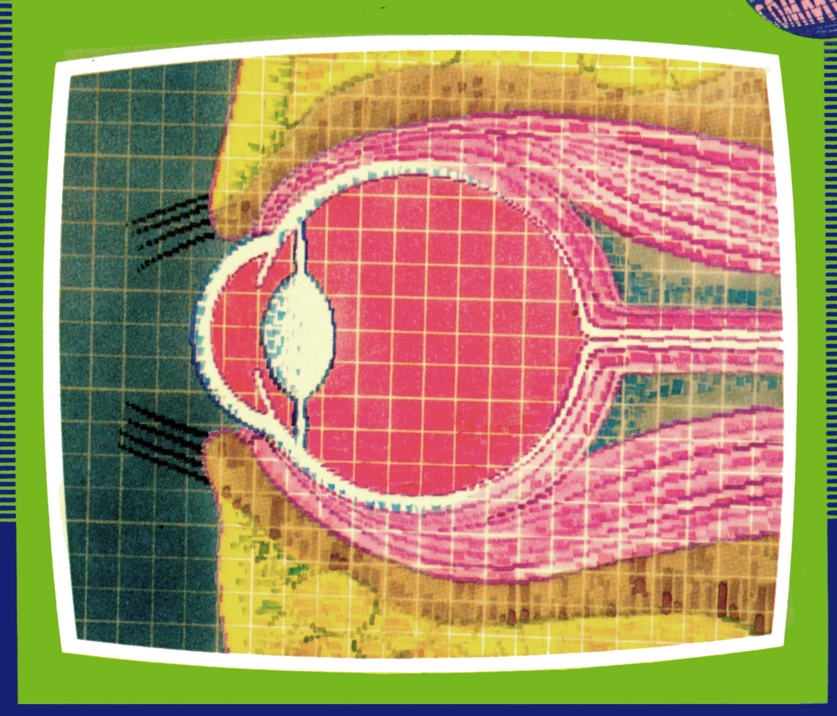
INFORMATICA APLICADA

27

Aplique su ordenador a las Ciencias Naturales

SINCLAIR
AMSTRAD
COMMODORE
IBM

S. Muñoz Martínez
F. P. Conde Montero-Ríos



EDICIONES SIGLO CULTURAL

ENCICLOPEDIA PRACTICA DE LA

INFORMATICA APLICADA

27

Aplique su ordenador a las Ciencias Naturales

EDICIONES SIGLO CULTURAL

Una publicación de

EDICIONES SIGLO CULTURAL, S.A.

Director-editor:
RICARDO ESPAÑOL CRESPO.

Gerente:
ANTONIO G. CUERPO.

Directora de producción:
MARIA LUISA SUAREZ PEREZ.

Directores de la colección:
MANUEL ALFONSECA, Doctor Ingeniero de Telecomunicación y Licenciado en Informática
JOSE ARTECHE, Ingeniero de Telecomunicación

Diseño y maquetación:
BRAVO-LOFISH.

Dibujos:
JOSE OCHOA Y ANTONIO PERERA.

Tomo XXVII. **Aplique su ordenador a las Ciencias Naturales.**
SATURNINO MUÑOZ MARTINEZ
FRANCISCO P. CONDE MONTERO-RIOS

Ediciones Siglo Cultural, S.A.

Dirección, redacción y administración:
Pedro Teixeira, 8, 2.ª planta (Iber a Mart-I). Teléf.: 810 52 13. 28020 Madrid.

Publicidad:
Gofar Publicidad, S.A. Benito de Castro, 12 bis. 28028 Madrid.

Distribución en España:
COEDIS, S.A. Valencia, 245. Teléf. 215 70 97. 08007 Barcelona.
Delegación en Madrid: Serrano, 165. Teléf. 411 11 48.

Distribución en Ecuador: Muñoz Hnos.

Distribución en Perú: DISELPESA.

Distribución en Chile: Alfa Ltda.

Importador exclusivo Cono Sur:
CADE, S.R.L. Pasaje Sud América. 1532. Teléf.: 21 24 64.
Buenos Aires - 1.290. Argentina.

ISBN del tomo: 84-7688-101-0
ISBN de la obra: 84-7688-018-9.

Fotocomposición:
ARTECOMP, S.A. Albarracín, 50. 28037 Madrid.

Imprime:
MATEU CROMO. Pinto (Madrid).

Depósito legal: M. 11.126-1987
Printed in Spain - Impreso en España.

Suscripciones y números atrasados:
Ediciones Siglo Cultural, S.A.
Pedro Teixeira, 8, 2.ª planta (Iberia Mart-I). Teléf.: 810 52 13. 28020 Madrid.

Mayo, 1987
P.V.P. Canarias: 365,-

INDICE

El número 27 me es fatal: si tuviera que probarlo diría que en día 27 nací. Doce veces al año amanece, sin embargo, día 27: soy supersticioso, porque el corazón del hombre necesita creer algo, y cree mentiras cuando no encuentra verdades que creer; sin duda, por esa razón creen los amantes, los casados y los pueblos a sus ídolos, a sus consortes y a sus gobiernos; y una de mis supersticiones consiste en creer que no puede haber para mí un día 27 bueno.

Don Mariano José de Larra, en
Artículos de Costumbres y Políticos.

DEDICATORIA

A los neófitos en el lenguaje científico.

A la hora de plantearse el hecho de escribir un libro, un fascículo o algo similar, lo primero que uno piensa es en el hecho de hacerlo bien, poniendo en ello toda su ciencia, sobre todo, en el caso de que el contenido no tenga que ver con los sentimientos, sino con la rama social a la que pertenece, en la que milita como ser productivo.

Después, al releer las diez primeras páginas, además de corregir el estilo, añadir algunas cosas, borrar muchas y no sentirse satisfecho con lo escrito, piensa en que está haciendo labor de divulgación, y en que no todo el mundo entiende su lenguaje deformado durante años de parquedad en vocablos; que no todo el mundo sabe qué es una proteína, aunque hayan oído hablar de ellas, el ADN, el ARN, los codones, etc. Entonces queda horrorizado: ¿cómo explicar algo que para uno es tan vital como el comer o el respirar, que forma parte de su entorno, de sus circunstancias, como dijo Ortega?

Después se calma, piensa que no es tan grave y que intentará ser lo más claro posible, explicar los términos más específicos y confiar en la buena fe del lector. Así calla a su conciencia.

Esta es nuestra intención: que los términos biológicos sean asequibles a aquellos que no son duchos en la materia, y que, por otra parte, si no es así, no acarreen demasiados problemas a las personas que traten de entender y utilizar estos programas.

Surge ahora otra duda: puede que nos encontremos en un caso opuesto —todo son complicaciones—, alguien que carezca de conocimientos de informática y al que los términos científicos le resulten básicos. También hemos tratado de pensar en ellos, pero aquí sólo pondremos buena voluntad, ya que confiamos en que, si ha entendido otros volúmenes de esta serie y le han servido en la lucha diaria con un ordenador, nuestros programas le serán comprensibles y útiles. Con el mejor propósito.

Si alguien pertenece a los que desconocen tanto el lenguaje científico como el informático, ¡el no va más!, suerte, amigo, y gracias por confiar en nosotros.

Hemos, pues, de completar la dedicatoria:

A los neófitos en el lenguaje científico y
a los neófitos en Informática.

Los autores.

INTRODUCCION



PARA que nadie se pregunte quiénes son Marga, Carlos y el doctor Ruiz, o de qué nos van a hablar, voy a contároslo en pocas palabras. El doctor Ruiz, al que todos llaman Fede, es el jefe de Marga y Carlos, que tratan de entrar en el mundillo de la ciencia. Carlos es biólogo, tiene veinticuatro años y pretende, según él mismo confiesa, trabajar en biología de ácidos nucleicos, ya que cree que en ellos está el futuro de la investigación biomédica. Margarita aún tiene que aprobar las asignaturas del último curso de la licenciatura en ciencias químicas, en su rama de bioquímica, ella también quiere dedicarse a esta actividad, porque le gusta y espera así poder hacer algo útil a los demás.

Marga y Carlos pertenecen a una generación de tránsito, en la que el ordenador está invadiendo todos los campos de la sociedad. No han tenido ocasión de jugar con ellos desde pequeños, de que fuera su petición a los Reyes Magos o un compañero insustituible en sus estudios. A ellos el ordenador personal les pilló adolescentes, casi universitarios, han vivido la explosión del Spectrum —cada vez más asequible— y hasta la informatización de sus facultades y laboratorios.

Al principio, recuerda Carlos, sólo pensaba en ellos como máquinas de marcianitos, simuladores de vuelo o expertos jugadores de ajedrez. Conocía, sólo de oídas, de sus aplicaciones a las Ciencias Naturales, pero es ahora, después de seis meses de tenerlos como compañeros de trabajo, cuando empieza a ver el gran número de posibilidades que ofrecen. A Marga le pasaba algo similar hasta que su hermano Juan decidió cursar la carrera de Informática y el ordenador se metió en casa.

Es difícil entender cómo funciona un ordenador, piensan ambos, no saben de sistema hexadecimal, octal o binario ni de la lógica enrevesada de los programas; pero tienen el gusanillo de la informática que crece cada vez que llegan al laboratorio y estos «aparatos» les ayudan y solucionan

sus problemas, les informan de las medidas de radiactividad, hacen un espectro de absorción o les ayudan a confeccionar las memorias de trabajo. Con el tiempo, serán capaces de manejarlos con soltura, ahora siempre andan con temor a perder los datos y todavía no saben que ese miedo les perseguirá siempre y muchas veces tendrán ganas de destruir el aparato que tienen delante, aunque sea tan necesario en su trabajo.

Fede entendió hace años que los ordenadores son amigos cada vez más útiles y hasta entretenidos; sabe mucho de software y de hardware, e intenta transmitir sus ideas a Marga y Carlos, ya que cree que un profesional de la investigación no puede prescindir de ellos. Quiere informatizar el laboratorio, que el trabajo cotidiano quede en la memoria privilegiada de los bytes y se pueda acceder a él rápida y cómodamente. Intenta entrar en los esquemas mentales de sus pupilos, que aprendan a utilizarlos de forma correcta: quizá lo consiga.

Este es un laboratorio en vías de informatización, no crean que eso hará su trabajo más automático, simplemente les ayudará y facilitará la labor. La creatividad la ponen ellos.

Todos los jueves, después del trabajo cotidiano, tienen una reunión informal sobre las aplicaciones de la informática a la Biología.

Los programas que aquí se recogen están hechos todos para el Spectrum de Sinclair, y en lenguaje Basic. Con ello, hemos pretendido que puedan ser de utilidad al mayor número de personas posible, ya que está llegando el momento en que, en todas las casas, este ordenador forma parte del ambiente familiar...

DIETETICA 1

Jueves, 13 de febrero de 1986

DESDE que hace un par de meses el doctor Ruiz sugirió la idea de dedicar cada semana unas horas a la informática; todos los jueves Marga y Carlos han seguido con mayor o menor entusiasmo las enseñanzas de su jefe. Algunos temas han resultado difíciles de entender pero, poco a poco, se han ido acostumbrando al lenguaje informático y a la forma en que se estructura un programa.

Hoy Fede les ha propuesto algo muy interesante:

CARLOS.—Estos seminarios deberían ser más tarde, después de comer no apetece hacer nada y mucho menos escuchar a alguien.

FEDE.—¡Siempre te andas quejando!, no sé cómo lo haces. Bueno, empecemos, precisamente hoy vamos a estudiar un programa relacionado con la alimentación; ¿alguno de vosotros sabe lo que es una caloría?

CARLOS.—Sí, lo que le entra a Marga cada vez que tiene un examen.

MARGA.—¡Qué gracioso!

CARLOS.—Bromas aparte, es una de las formas de medir la energía producida por la utilización de los diversos alimentos. En metabolismo se utiliza la caloría grande o kilocaloría, que puede definirse como:

> *Caloría grande o Cal es la cantidad de calor necesaria para elevar la temperatura de un kilogramo de agua de 14,5° a 15,5° C.*

Este concepto es muy utilizado en medicina al ser índice del metabolismo humano; así se estudian los alimentos como fuente externa de energía por la cantidad de calorías que proporcionan cuando son digeridos en el organismo y esto permite confeccionar una dieta equilibrada, en la que no falte nada de lo necesario para el buen funcionamiento de nuestro cuerpo.

FEDE.—¡Vaya parrafada!, se nota que te interesa el tema.

CARLOS.—Claro, es casi una cuestión personal.

FEDE.—¿Por qué?

CARLOS.—Cuando era pequeño, estaba muy gordo, mi madre empezó a preocuparse y decidió llevarme al médico. Este dijo que si las cosas no cambiaban al hacerme mayor, pensaría en una dieta o en hacerme el metabolismo basal.

MARGA.—Perdona Carlos, siempre he oído esa expresión de hacerse el metabolismo y nunca la entendí bien.

CARLOS.—Es fácil, existe algo que los fisiólogos llaman metabolismo basal, esto es la cantidad mínima de calor producida por una persona en ayunas a una temperatura media de 20° C, y supone la energía necesaria para mantener las funciones vegetativas de una forma normal, tales como la respiración, circulación sanguínea, temperatura del cuerpo, etc. Si estas necesidades no son cubiertas o se hace de forma excesiva, ocasionan la desnutrición o la obesidad exagerada como casos extremos.

FEDE.—La medida del metabolismo basal se tiene que hacer en condiciones determinadas: de diez a catorce horas después de haber comido y con el sujeto en reposo física y emocionalmente.

MARGA.—¿Y por qué puede verse alterado este metabolismo basal?

CARLOS.—Por diferentes causas que van desde una alimentación inadecuada hasta una enfermedad metabólica de tipo enzimático. Si se estudia con detenimiento, puede llegarse a una solución con la administración de una dieta equilibrada o con la eliminación del alimento que ocasiona el trastorno.

FEDE.—Es cierto, hay muchas enfermedades que son provocadas por la falta o mal funcionamiento de algunos sistemas enzimáticos, que hacen al organismo intolerante a algún metabolito. Así están la intolerancia al gluten de germen de trigo o a la lactosa; estas enfermedades pueden ser solventadas con la eliminación de los alimentos que lleven dichas sustancias.

MARGA.—Ya voy entendiendo la importancia de una buena alimentación. ¿Sabes cuántas calorías necesita una persona cada día?

CARLOS.—No exactamente.

FEDE.—Yo te puedo responder, Marga: se calcula que para un varón de veinticinco años, 70 kg. de peso y que viva en un clima templado de aproximadamente 20° C, son necesarias 45 kilocalorías por kilogramo de peso y día.

MARGA.—¿Y no dice nada del color de los ojos, si está soltero y es guapo?

FEDE.—No, lo que pasa es que las necesidades calóricas cambian dependiendo de muchas variables: sexo, edad, clima, tipo de actividad, etc.; por ejemplo, en las mismas condiciones, una mujer sólo necesita unas 35 kilocalorías.

CARLOS.—Por eso engordan antes...

FEDE.—Déjalo ya, Carlos. Un niño tiene necesidades metabólicas más altas, ya que sus tejidos crecen mucho más rápidamente; así, por ejemplo, el lactante necesita unas 110 kilocalorías por kilogramo de peso y día; de

uno a tres años son necesarias 100, de cuatro a seis años, 90, que es casi dos veces las necesidades de un adulto. Por otra parte, un niño de tres años consume el 50 por ciento de sus calorías en mantener su metabolismo basal, mientras que en un adulto ese porcentaje baja al 30 por ciento. A medida que se envejece se necesita menos energía; a los sesenta y cinco años sólo son necesarias 36 kilocalorías en los hombres y 27 en las mujeres.

Como sabéis, el cuerpo humano ha de mantener una temperatura idónea, de aproximadamente 37° C. Por ello, los habitantes de zonas frías tienen un metabolismo basal más elevado, ya que han de protegerse del frío, y en el Ecuador ocurre lo contrario, no se necesita consumir tanta energía para mantener esta temperatura corporal.

MARGA.—Ya veo, a lo largo de la vida se pasa por muchos estados en los que el organismo necesita diferente cantidad de alimentos.

CARLOS.—Tengo curiosidad por saber cómo se confecciona una dieta, qué cantidad de calorías se mantienen, si es necesario incluir algún medicamento o algo así.

FEDE.—Hoy por hoy esto constituye una rama de la medicina que está adquiriendo gran importancia. Son varias las especialidades médicas que se ocupan de este tema, así que es muy difícil generalizar; cada situación es algo particular y sólo un especialista puede valorarla.

Al elaborar una dieta es necesario tener en cuenta muchos factores, algunos curiosos, como la cantidad de fibras, el factor de digestibilidad de los alimentos, la capacidad de saciedad que tienen, la capacidad de síntesis de vitaminas de las bacterias intestinales, diversos factores endocrinos, factores emocionales, y hasta la radiactividad...

MARGA.—Especialmente después del accidente de Chernobil.

FEDE.—No sólo por eso, la radiactividad de la central que llegó a España parece ser prácticamente nula.

MARGA.—Por curiosidad, ¿es cierto eso de que, durante el embarazo la mujer ha de comer por dos?

FEDE.—Nada de eso, en la segunda mitad del proceso de gestación es necesaria una alimentación más rica, pero tan sólo de 300 kilocalorías al día; sin embargo, este valor se multiplica por tres durante la lactancia.

CARLOS.—La leche parece un producto caro para el organismo.

FEDE.—Cierto, ten en cuenta que ha de ser muy rica en hidratos de carbono, lípidos y proteínas, así como las vitaminas y minerales necesarios para el lactante.

CARLOS.—Otra cosa, hablas siempre de kilocalorías, ¿por qué?

FEDE.—Pues porque las medidas que se dan en dietética se refieren a 1.000 unidades de calor; el hombre, en realidad, necesita 45.000 unidades cada día por kilogramo de peso. Se ha generalizado el concepto de Caloría, escrito con mayúscula, por ser más cómodo, pero equivale a una kilocaloría.

MARGA.—Como sigamos así, no vamos a acabar nunca de hablar de me-

tabolismo, y os recuerdo que aún no hemos empezado con el programa de hoy.

```
   1 REM Principal
   2 REM
  10 GO SUB 100
  30 GO SUB 500
  40 GO SUB 2000
  50 INPUT "mas productos (s,n):
";m$
  60 IF m$<>"n" THEN GO TO 10
  70 GO TO 9999
 100 REM entrada de datos
 110 REM
 120 INPUT "Producto: ";p$: GO S
UB 1900
 130 INPUT "Cantidad en gramos:"
;cantidad
 140 RETURN
 500 REM calculos
 510 REM
 520 LET calorias=(cantidad/100)
*cal0
 530 RETURN
1000 REM Tabla de alimentos
1010 REM
1020 DATA "aceituna",75,1.5,13.5
,0,4.0,1.2,132,300,0,0,0,0
1021 DATA "aguacate",65,1.7,26.4
,0,5.1,1.8,245,290,0.06,0.13,1.1
,16
1022 DATA "albaricoque",85.3,0.9
,0.2,0.003,12.9,0.6,51,2790,0.03
,0.04,0.7,8
1023 DATA "ananas",86.7,0.4,0.2,
0,12.2,0.5,47,130,0.08,0.02,0.2,
24
1024 DATA "arandano",83.4,0.6,0.
6,0,15.1,1.2,61,280,0.03,0.07,0.
3,16
1025 DATA "cereza",83.4,1.1,0.4,
0,14.6,0.5,60,1000,0.05,0.06,0.1
4,8
1026 DATA "ciruela",85.7,0.7,0.2
,0,12.9,0.5,50,350,0.15,0.03,0.6
,5
1027 DATA "frambuesa",83,1.1,0.6
,0,14.4,2.8,66,150,0.03,0.07,0.3
,25
1028 DATA "fresa",90,0.8,0.6,0.0
09,8.1,1.2,37,60,0.03,0.07,0.3,6
0
1029 DATA "grosella",83.7,1.4,0.
4,0,13.9,3.2,58,120,0.05,0.14,0.
3,35
1030 DATA "higo",71.7,1.2,0.4,0.
01,16.1,1.4,65,75,0.09,0.08,0.63
,2
1031 DATA "limon",89.3,0.9,0.6,0
,8.7,0.9,32,0,0.04,0,0.1,45
1032 DATA "mandarina",87,0.8,0.3
,0,10.9,1.0,44,420,0.07,0.03,0.2
,31
1033 DATA "manzana",84,0.3,0.4,0
.007,15,0.9,58,90,0.04,0.02,0.2,
16
1034 DATA "melocoton",86.6,0.8,0
.2,0.006,11.8,0.6,47,880,0.02,0.
05,0.9,8.7
1035 DATA "melon",94,0.6,0.2,0,4
.6,0.6,20,3420,0.05,0.04,0.5,33
1036 DATA "membrillo",84,0.3,0,0
,14.1,0,52,0,0,0,0,10
```

```
1037 DATA "mora",84,1.2,1.1,0,11
.9,4.1,58,200,0.03,0.04,0.31,21
1038 DATA "naranja",87.1,0.9,0.2
,0.014,11.3,0.8,45,190,0.08,0.03
,0.2,49
1039 DATA "pera",83.2,0.5,0.4,0.
0014,15.5,1.5,61,20,0.02,0.04,0.
1,4
1040 DATA "platano",73.5,1.3,0.4
,0.008,24,0.5,94,430,0.09,0.06,0
.6,10
1041 DATA "pina",86.7,0.4,0.2,0,
12.2,0.5,47,130,0.08,0.02,0.2,24
1042 DATA "pomelo",89,0.6,0.2,0,
9.8,0.5,39,0,0.04,0.02,0.2,40
1043 DATA "sandia",92.1,0.5,0.2,
0,017,6.9,0.6,28,590,0.05,0.05,0
.2,6
1044 DATA "toronja",89,0.6,0.2,0
,9.8,0.5,39,0,0.04,0.02,0.2,40
1045 DATA "uva",81.6,0.8,0.4,0.0
06,16.7,4.3,68,80,0.05,0.03,0.4,
4
1900 REM Busca en tabla
1910 REM
1915 RESTORE 1020
1920 FOR i=1 TO 26
1930 READ r$,agua,prot,liptot,co
lest,carbotot,carbofib,cal0,vitA
,vitB1,vitB2,acnicot,vitC
1940 IF p$=r$ THEN RETURN
1950 NEXT i
1960 BEEP 0.05,28: PRINT "No enc
uentro este producto"
1970 GO TO 10
2000 REM imprime
2010 REM
2020 INPUT "Pantalla o Impresora
(p o i): ";t$
2030 IF t$<>"p" AND t$<>"i" THEN
 GO TO 2020
2040 LET ca=(2 AND t$="p")+(3 AN
D t$="i")
2050 PRINT cantidad;" g de ";p$;
" tienen ";INT (calorias*10)/10'
"Calorias"
2060 RETURN
```

FEDE.—Tienes razón, vamos con ello.

Este es el esqueleto de un programa que puede hacer muchas cosas. Está elaborado para el Spectrum y tiene varias rutinas que pueden ser modificadas por vosotros dependiendo de qué queráis hacer. Mi idea es que los programas que vayamos haciendo sean flexibles y los podáis adaptar a la hora de necesitar resolver problemas concretos.

```
  1 REM Principal
  2 REM
 10 GO SUB 100
 30 GO SUB 500
 40 GO SUB 2000
 50 INPUT "mas productos (s,n):
";m$
 60 IF m$<>"n" THEN GO TO 10
 70 GO TO 9999
```

Desde la línea 10 a la 70 está lo que en Pascal se conoce como rutina *principal* (MAIN). Esta rutina, que ejecuta el resto del programa, consiste fundamentalmente en una serie de GOSUB seguidos de un número de línea. Así, la línea 10 tiene GOSUB 100 que ejecuta la rutina *entrada de datos* que termina con RETURN (vuelve, en inglés). El RETURN vuelve a la línea 10 pero a los : que hay detrás de GOSUB 100, para ejecutar a continuación el REM, que es como decirle al computador: «lo que está detrás de REM es un secreto mío y a ti no te interesa». El ordenador, siempre muy obediente, no lo tiene en cuenta y ejecuta a continuacióon la línea 30. De esta manera se van ejecutando las líneas 10 a 40 para, al llegar a la 50, preguntarnos si queremos saber las calorías de más productos. Si al INPUT de esta línea respondemos: s, el programa se vuelve a ejecutar desde el principio, ya que vamos a la línea 10. Pero si respondemos: *n*, la ejecución continúa en la línea 9999, donde termina al no haber más líneas detrás.

```
 100 REM entrada de datos
 110 REM
 120 INPUT "Producto: ";p$: GO S
UB 1900
 130 INPUT "Cantidad en gramos:"
;cantidad
 140 RETURN
```

Como comentábamos, la primera rutina que se ejecuta es la que empieza en la línea 100, *entrada de datos*. Las dos primeras líneas, 100 y 110, nos sirven a nosotros para saber dónde empiezan y terminan las rutinas con un solo vistazo al programa. La línea 120 tiene un INPUT que pregunta el producto cuyas calorías queremos buscar introduciendo nuestra contestación en la variable p$. En la línea 130 se hace lo mismo para la cantidad en gramos de este producto del que queremos saber las calorías. La rutina termina con RETURN en la línea 140.

```
1900 REM Busca en tabla
1910 REM
1915 RESTORE 1020
1920 FOR i=1 TO 26
1930 READ r$,agua,prot,liptot,co
lest,carbotot,carbofib,cal0,vitA
,vitB1,vitB2,acnicot,vitC
1940 IF p$=r$ THEN RETURN
1950 NEXT i
1960 BEEP 0.05,28: PRINT "No enc
uentro este producto"
1970 GO TO 10
```

La segunda rutina ejecutada es *búsqueda en la tabla*. Esta tabla está entre las líneas 1000 y 1900 y la comentaremos después. La rutina de búsqueda empieza en 1915 con RESTORE 1020, que consigue que el READ (lee, en inglés) empiece a leer siempre en el DATA de la línea 1020. Cuan-

do se ejecuta esta rutina por primera vez el READ no es estrictamente necesario pero si solamente leemos un fragmento de la tabla, la siguiente vez que lo intentemos, nos encontraremos que el computador empezaría a continuación de donde hubiese leído la primera y, aparte de no encontrar el dato que buscábamos, si está antes, cometería un error al no tener la tabla datos suficientes para ejecutar el lazo FOR TO las veces indicadas.

El lazo, líneas 1920 a 1950, busca los datos en la tabla leyendo los valores por 100 gramos de producto e introduciéndolos en las variables r$ y agua = cantidad de agua, prot = proteínas, liptot = lípidos totales, colest = colesterol, carbotot = hidratos de carbono, carbofib = fibras, cal0 = calorías, vitA = vitamina A, vitB1 = vitamina B_1, vitB2 = vitamina B_2, acnicot = ácido nicotínico y vitC = vitamina C por cada 100 gramos. A continuación el IF de la línea 1940 comprueba que el producto en r$ es el que buscábamos y que estaba en p$. Si lo es, termina con RETURN, que vuelve a la línea 20 detrás del GOSUB, y si no lo es continúa en la línea siguiente, que aumenta 1 a la variable i; NEXT i.

Si el lazo se ejecuta y el producto que buscamos no está en la tabla o está mal escrito, el programa llega a la línea 1960 y después de emitir un sonido, ejecutando BEEP, escribe en la pantalla un letrero que avisa de que no tiene datos de este producto. La rutina termina en la línea siguiente con GOTO 10.

```
1000 REM Tabla de alimentos
1010 REM
1020 DATA "aceituna",75,1.5,13.5
,0,4.0,1.2,132,300,0,0,0,0
1021 DATA "aguacate",65,1.7,26.4
,0,5.1,1.8,245,290,0.06,0.13,1.1
,16
1022 DATA "albaricoque",85.3,0.9
,0.2,0.003,12.9,0.6,51,2790,0.03
,0.04,0.7,8
1023 DATA "ananas",86.7,0.4,0.2,
0,12.2,0.5,47,130,0.08,0.02,0.2,
24
1024 DATA "arandano",83.4,0.6,0.
6,0,15.1,1.2,61,280,0.03,0.07,0.
3,16
1025 DATA "cereza",83.4,1.1,0.4,
0,14.6,0.5,60,1000,0.05,0.06,0.1
4,8
1026 DATA "ciruela",85.7,0.7,0.2
,0,12.9,0.5,50,350,0.15,0.03,0.6
,5
1027 DATA "frambuesa",83,1.1,0.6
,0,14.4,2.8,66,150,0.03,0.07,0.3
,25
1028 DATA "fresa",90,0.8,0.6,0.0
09,8.1,1.2,37,60,0.03,0.07,0.3,6
0
1029 DATA "grosella",83.7,1.4,0.
4,0,13.9,3.2,58,120,0.05,0.14,0.
3,35
1030 DATA "higo",71.7,1.2,0.4,0.
01,16.1,1.4,65,75,0.09,0.08,0.63
,2
1031 DATA "limon",89.3,0.9,0.6,0
,8.7,0.9,32,0,0.04,0,0.1,45
```

```
1032 DATA "mandarina",87,0.8,0.3
,0,10.9,1.0,44,420,0.07,0.03,0.2
,31
1033 DATA "manzana",84,0.3,0.4,0
.007,15,0.9,58,90,0.04,0.02,0.2,
16
1034 DATA "melocoton",86.6,0.8,0
.2,0.006,11.8,0.6,47,880,0.02,0.
05,0.9,8.7
1035 DATA "melon",94,0.6,0.2,0,4
.6,0.6,20,3420,0.05,0.04,0.5,33
1036 DATA "membrillo",84,0.3,0,0
,14.1,0,52,0,0,0,0,10
1037 DATA "mora",84,1.2,1.1,0,11
.9,4.1,56,200,0.03,0.04,0.31,21
1038 DATA "naranja",87.1,0.9,0.2
,0.014,11.3,0.8,45,190,0.08,0.03
,0.2,49
1039 DATA "pera",83.2,0.5,0.4,0.
0014,15.5,1.5,61,20,0.02,0.04,0.
1,4
1040 DATA "platano",73.5,1.3,0.4
,0.008,24,0.5,94,430,0.09,0.06,0
.6,10
1041 DATA "pina",86.7,0.4,0.2,0,
12.2,0.5,47,130,0.08,0.02,0.2,24
1042 DATA "pomelo",89,0.6,0.2,0,
9.8,0.5,39,0,0.04,0.02,0.2,40
1043 DATA "sandia",92.1,0.5,0.2,
0.017,6.9,0.6,28,590,0.05,0.05,0
.2,6
1044 DATA "toronja",89,0.6,0.2,0
,9.8,0.5,39,0,0.04,0.02,0.2,40
1045 DATA "uva",81.6,0.8,0.4,0.0
06,16.7,4.3,68,80,0.05,0.03,0.4,
4
```

La tabla de datos que he puesto corresponde a la de frutas frescas y como podéis apreciar el contenido de vitamina C es máximo en la fresa, aunque popularmente se considera la naranja como el alimento más rico en esta vitamina.

Un momento... creo que antes deberíamos hablar un poco de las vitaminas y minerales que es necesario incluir en la dieta. Imagino que tú, Marga, habrás traído las paginitas que quedamos que escribirías sobre esto...

MARGA.—Sí, las leo...

Además de los componentes mayoritarios los seres vivos necesitan otros en cantidades mínimas, estos son las vitaminas y los minerales. Estas sustancias son vitales para muchas formas de vida y su gran importancia biológica es debida a que algunos organismos no las pueden sintetizar y necesitan adquirirlas con los alimentos.

Que algunas enfermedades están relacionadas con la carencia en la dieta de alguno de estos compuestos es algo conocido desde la antigüedad; así las madres de Terranova daban a sus hijos aceite de hígado de bacalao para que creciesen bien. Hoy en día se sabe que éste es uno de los alimentos más ricos en vitamina D, por lo que previene el raquitismo.

En el año 1912, Funk obtuvo un concentrado de una amina de la cascarilla de arroz que aliviaba los síntomas del *beri-beri*, a esta sustancia le

llamó *vitamina* o amina esencial para la vida. Desde entonces se utiliza este término para definir sustancias de acción similar.

A partir de 1935 se dio una búsqueda sistemática de vitaminas partiendo, a veces, de toneladas de material para obtener unos cuantos miligramos. En pocos años se describieron muchas de las vitaminas que ahora conocemos.

La determinación de las cantidades precisas de vitamina procedente de los alimentos que necesita el hombre es siempre difícil de valorar debido a que parte o toda la demanda de las mismas puede obtenerse gracias a que ciertas bacterias, normalmente presentes en el intestino, pueden sintetizarlas.

He elaborado una tabla en donde se recogen las necesidades aproximadas de vitaminas para el hombre:

Necesidades vitamínicas diarias

	Vitaminas			
	U.I.[1]	miligramos		
Edad	**A**	**B_1**	**B_2**	**C**
Lactantes	1.500	0,4	0,6	30
De 1 a 3 años	2.000	0,6	0,9	40
De 4 a 6 años	2.500	0,8	1,2	50
De 7 a 9 años	3.500	1,0	1,5	60
Adultos	5.000	1,6	2,2	90

[1] U.I. = 0,0006 mg. beta-caroteno.

FEDE.—¿De dónde sacaste los datos?

MARGA.—De varios libros, jefe, son bastante fiables aunque están redondeados y de entre las cifras que he encontrado, he escogido los que me gustaron más.

FEDE.—Bueno..., pase..., pero yo no me fiaría mucho de tus datos. Cuando se escribe una tabla como ésta se pone una de un organismo fiable diciendo de qué organismo es; eso de refundir datos de este tipo es peligroso.

Empezaste a hablar de las cantidades de vitaminas que tienen algunos alimentos y nos has dejado a medias.

MARGA.—Puedo seguir haciéndolo si me dejáis.

Es curioso constatar que la mandarina contiene bastante menos vitamina C que la naranja o el limón. El higo tiene un contenido muy bajo, así como la pera. La deficiencia de vitamina C, que provoca el escorbuto en el hombre, casi no se ve en los animales superiores, ya que son capaces de sintetizarla, cosa que nosotros no podemos hacer.

Si hablamos de contenido en vitamina A cada unidad representa

0,0006 mg. de beta-caroteno; es curioso constatar que el melón tiene 3.420 unidades por 100 gramos, el albaricoque 2.790 y, en cambio, por ejemplo, la manzana sólo tiene 90 y la naranja 100. La famosa zanahoria contiene entre 2.000 y 12.000 unidades de vitamina A por 100 gramos.

La falta de vitamina A provoca enfermedades como la *xeroftalmía* (ojos secos), que consitituye un síntoma temprano, así como la ceguera nocturna, que consiste en la dificultad de adaptación a la oscuridad y se utiliza como prueba para diagnóstico. Sin embargo, el hígado de los adultos puede almacenar suficiente vitamina A para varios años, por lo que su deficiencia, que es una causa corriente de ceguera en algunas zonas tropicales donde la alimentación es escasa, se produce más frecuentemente en los jóvenes.

Las vitaminas del grupo B son más corrientes en las carnes, volatería, pescados y frutos secos. De entre ellas la vitamina B_1, «pirofosfato de tiamina», es necesaria en la dieta de la mayor parte de los vertebrados y su deficiencia produce el beri-beri en el hombre. Cuando se retira la vitamina B_1 de la dieta en animales de experimentación, los tejidos sufren su deficiencia progresivamente y al ser afectados desigualmente, nos permiten reconocer el *beri-beri*.

De las frutas recogidas en la tabla la que mayor contenido en vitamina B_1 tiene es la ciruela fresca, con 0,15 mg. por 100 gramos, mientras que en almíbar sólo contiene 0,03 mg., y seca 0,1 mg. Entre los componentes inorgánicos de las frutas hay que destacar que el plátano tiene un alto contenido en potasio: 420 mg. por 100 gramos, también tienen una cantidad elevada el albaricoque, 440 mg.; el aguacate, con 340 mg., o las uvas secas, con 708 mg.

El calcio aparece en gran cantidad en la aceituna, con 87 mg.; en la frambuesa, con 49 mg., entre las frutas; pero la almendra tiene 254 mg., la avellana 290 mg., la leche de vaca 125 mg. y la leche en polvo entera 949 mg., y entre los quesos, el de Parma, 1.220 mg., y el Emmental, 1.090 mg. Entre las hortalizas se llevan la palma el berro, con 187 mg., y el perejil, con 190 mg.

El hierro, otro oligoelemento que le interesa mucho a las madres de familia, no aparece en gran cantidad en la fruta, sólo la frambuesa, la fresa, el arándano y el higo tienen alrededor de 1 mg. por 100 g., mientras que el berro tiene 2 mg., la espinaca fresca, 3 mg., el guisante fresco, 2 mg., la judía blanca seca, 10,3 mg., y la lenteja seca, 8,6 mg. por 100 g., aunque parezca mentira, debido a la fama de la lenteja de tener mucho hierro.

CARLOS.—¿El hígado tiene mucho hierro?

MARGA.—Sí, el hígado de cerdo tiene 18 mg. por 100 g., también yo me hice esta pregunta.

Por último, el fósforo es abundante en frutos secos; así, por ejemplo, la almendra tiene 475 mg. por 100 g. y la nuez de Brasil 600 mg., alrededor de 400 mg. la avellana, el cacahuete y la nuez. Sin embargo, otros ali-

mentos con alto contenido en fósforo son el germen de trigo, con 1.096 mg. por 100 g., o el salvado de arroz, con 1.450 mg., y la lenteja seca, con 430 mg., el chocolate, con 400 mg., o los quesos, con unos 600 mg.

Entre los alimentos que tienen más calorías se encuentran las grasas y aceites, con 900 cal. por 100 g. para la manteca de cerdo y el aceite de hígado de bacalao, el aceite de oliva, con 880 cal. Hay que hacer notar que nuestras madres tenían razón cuando nos daban aceite de hígado de bacalao, porque contiene 85.000 unidades de vitamina A y 10.000 de vitamina D por 100 g. Otros alimentos con muchas calorías son la almendra, la avellana, el cacahuete y la nuez, de 670 a 560 cal. por 100 g. El chocolate tiene 570 cal. por 100 g., la leche en polvo 496 cal., el pan tostado 422 cal., y el pan blanco 260 cal. Entre las frutas que tienen más calorías por 100 gramos están la uva, la frambuesa, el higo y el plátano; el melón, la sandía, el limón y la fresa son las que menos. Esta tabla se puede completar fácilmente para todos los alimentos con un libro de dietética.

```
500 REM calculos
510 REM
520 LET calorias=(cantidad/100)
*cal0
530 RETURN
```

FEDE.—Sigo...

El programa continúa ejecutando la rutina *cálculo de calorías,* que empieza en la línea 500. En ella, línea 520, se hace una sencilla regla de tres para calcular las calorías en los gramos del producto que queramos. La rutina termina en la línea 530 con RETURN.

Una vez ejecutada *cálculos de calorías,* en la variable numérica calorías tendremos la respuesta buscada del número de calorías para la cantidad dada de un alimento y la línea 40 ejecuta *imprime,* que empieza en la línea 2000 y que en su línea 2020 pregunta si se quiere imprimir en la pantalla o en una impresora; la respuesta, *p* o *i,* se comprueba en la línea 2030, que en caso de no ser ninguna de las dos vuelve a la 2020. La línea 2040 genera una variable *Ca* que contendrá 2 en el caso de que la respuesta sea *p* y 3 en el caso de que sea *i.* La línea 2050 imprime en la pantalla o en la impresora el resultado (PRINT #ca;). La rutina y el programa terminan en la línea siguiente, 2060.

```
2000 REM imprime
2010 REM
2020 INPUT "Pantalla o Impresora
 (p o i): ";t$
2030 IF t$<>"p" AND t$<>"i" THEN
 GO TO 2020
2040 LET ca=(2 AND t$="p")+(3 AN
D t$="i")
```

```
2050 PRINT cantidad;" g de ";p$;
" tienen ";INT (calorias*10)/10
"Calorias"
2060 RETURN
```

Una vez explicado el programa os pediré variaciones. En primer lugar, y empezando por lo más fácil, variando o aumentando los DATA podemos incluir nuevos alimentos, variar los que ya tenemos o poner otros datos. Para tomarse el programa en serio e introducir una tabla completa de alimentos habría primero que convertirlo en algo útil.

De todas maneras, Marga, ¿qué líneas habría que variar para que este programa admitiese 50 líneas más con un DATA cada una?

MARGA.—Habría, en primer lugar, que poner los DATA de la misma manera a los que están en las líneas 1020 a 1045. Después... creo que... lo único necesario sería modificar la línea 1920 para que el lazo revisase los DATA nuevos.

FEDE.—Bien, pero si en lugar de 50 DATA más empiezas un sábado a meter DATA y después de dos horas estás hasta el gorro de DATA, ¿cómo sabrías cuántos DATA tendrías en el programa sin necesidad de contarlos?

MARGA.—Lo más fácil es modificar el lazo así:

FOR i = 1 TO 1000

escribir LET p$=" " y ejecutar el programa a partir de la línea 1915 escribiendo GOTO 1915 y ENTER. El programa ejecutado de esta manera cometería un error «out of data at line 1930». Si ahora escribimos PRINT i y ENTER, y al valor que nos da en la pantalla le restamos 1, es el número de DATA que hay en el programa. Este valor es el que hay que poner en el lazo FOR TO.



```
Cantidad en gramos:241
```

```
Pantalla o Impresora (p o i):p
```

```
241 g de cereza tienen 144.6
Calorias

mas productos (s,n):'
```





FEDE.—Perfecto, ahora vamos a pasar a algo más complicadillo.

Vamos a cambiar el programita de manera que podamos obtener tablas de los alimentos ordenados dependiendo de su contenido en algún componente:

```
   1 REM alimentos 2
   2 REM
   5 PRINT AT 10,0; INVERSE 1;"P
or favor espere, tengo trabajo"
   9 GO SUB 5000
  10 GO SUB 1900
  20 CLS : GO SUB 4000
  30 GO SUB 2000
```

```
  40 CLS
  50 GO SUB 3000
  70 INPUT "mas productos (s,n):
 ";m$
  80 IF m$<>"n" THEN CLS : FOR s
=1 TO 10: PRINT #ca: NEXT s: GO
TO 20
  90 GO TO 9999
1000 REM Tabla de alimentos
1010 REM
1020 DATA "aceituna",75,1.5,13.5
,0,4.0,1.2,132,300,0,0,0,0
1021 DATA "aguacate",65,1.7,26.4
,0,5.1,1.8,245,290,0.06,0.13,1.1
,16
1022 DATA "albaricoque",85.3,0.9
,0.2,0.003,12.9,0.6,51,2790,0.03
,0.04,0.7,8
1023 DATA "ananas",86.7,0.4,0.2,
0,12.2,0.5,47,130,0.08,0.02,0.2,
24
1024 DATA "arandano",83.4,0.6,0.
6,0,15.1,1.2,61,280,0.03,0.07,0.
3,16
1025 DATA "cereza",83.4,1.1,0.4,
0,14.6,0.5,60,1000,0.05,0.06,0.1
4,8
1026 DATA "ciruela",85.7,0.7,0.2
,0,12.9,0.5,50,350,0.15,0.03,0.6
,5
1027 DATA "frambuesa",83,1.1,0.6
,0,14.4,2.8,66,150,0.03,0.07,0.3
,25
1028 DATA "fresa",90,0.8,0.6,0.0
09,8.1,1.2,37,60,0.03,0.07,0.3,6
0
1029 DATA "grosella",83.7,1.4,0.
4,0,13.9,3.2,58,120,0.05,0.14,0.
3,35
1030 DATA "higo",71.7,1.2,0.4,0.
01,16.1,1.4,65,75,0.09,0.08,0.63
,2
1031 DATA "limon",89.3,0.9,0.6,0
,8.7,0.9,32,0,0.04,0,0.1,45
1032 DATA "mandarina",87,0.8,0.3
,0,10.9,1.0,44,420,0.07,0.03,0.2
,31
1033 DATA "manzana",84,0.3,0.4,0
.007,15,0.9,58,90,0.04,0.02,0.2,
16
1034 DATA "melocoton",86.6,0.3,0
.2,0.006,11.8,0.6,47,880,0.02,0.
05,0.9,8.7
1035 DATA "melon",94,0.6,0.2,0,4
.6,0.6,20,3420,0.05,0.04,0.5,33
1036 DATA "membrillo",84,0.3,0,0
,14.1,0,52,0,0,0,0,10
1037 DATA "mora",84,1.2,1.1,0,11
.9,4.1,56,200,0.03,0.04,0.31,21
1038 DATA "naranja",87.1,0.9,0.2
,0.014,11.3,0.8,45,190,0.08,0.03
,0.2,49
1039 DATA "pera",83.2,0.5,0.4,0.
0014,16.5,1.5,61,20,0.02,0.04,0.
1,4
1040 DATA "platano",73.5,1.3,0.4
,0.008,24,0.5,94,430,0.09,0.06,0
.6,10
1041 DATA "pina",86.7,0.4,0.2,0,
12.2,0.5,47,130,0.08,0.02,0.2,24
1042 DATA "pomelo",89,0.6,0.2,0,
9.8,0.5,39,0,0.04,0.02,0.2,40
1043 DATA "sandia",92.1,0.5,0.2,
0.017,6.9,0.6,28,590,0.05,0.05,0
.2,6
```

```
1044 DATA "toronja",89,0.6,0.2,0
,9.8,0.5,39,0,0.04,0.02,0.2,40
1045 DATA "uva",81.6,0.8,0.4,0.0
06,16.7,4.3,68,80,0.05,0.03,0.4,
4
1900 REM Busca en tabla
1910 REM
1915 RESTORE 1020: DIM r$(26,12)
: DIM n(12,26)
1920 FOR i=1 TO 26
1930 READ r$(i): FOR s=1 TO 12:
READ n(s,i): NEXT s
1940 NEXT i
1950 RETURN
2000 REM pantalla impresora
2010 REM
2020 INPUT "Pantalla o Impresora
(p o i): ";t$
2030 IF t$<>"p" AND t$<>"i" THEN
 GO TO 2020
2040 LET ca=(2 AND t$="p")+(3 AN
D t$="i")
2060 RETURN
3000 REM ordena
3010 REM
3020 PRINT AT 10,10;"Trabajo": D
IM w(26): DIM z(26)
3030 FOR s=1 TO 26: LET z(s)=n(l
,s)-s*(n(l,s)=0): NEXT s
3040 FOR t=1 TO 26: LET max=z(t)
3050 FOR s=1 TO 26
3070 IF z(s)>=max THEN LET max=z
(s): LET w(t)=s
3080 NEXT s
3090 LET z(w(t))=-999
3100 NEXT t
3110 CLS : PRINT #ca;l;" = ";l$(
l)'': FOR s=1 TO 26
3130 LET y=w(s): PRINT #ca;(" "
AND s<10);s;" ";r$(y);" ";n(l,y)
3140 NEXT s
3150 RETURN
4000 REM menu
4010 REM
4020 PRINT AT 0,1;"***CONTENIDO
EN 100 GRAMOS***";AT 2,11;"de FR
UTA"
4030 PRINT AT 7,0;" 1 Contenido
en Agua"
4040 PRINT AT 8,0;" 2 Contenido
en Proteinas"
4050 PRINT AT 9,0;" 3 Contenido
en Lipidos totales"
4060 PRINT AT 10,0;" 4 Contenido
 en Colesterol"
4070 PRINT AT 11,0;" 5 Contenido
en Hidratos Carbono"
4080 PRINT AT 12,0;" 6 Contenido
en Fibras"
4090 PRINT AT 13,0;" 7 Contenido
en Calorias"
4100 PRINT AT 14,0;" 8 Contenido
en Vitamina A"
4110 PRINT AT 15,0;" 9 Contenido
en Vitamina B1"
4120 PRINT AT 16,0;"10 Contenido
en Vitamina B2"
4130 PRINT AT 17,0;"11 Contenido
en Acido Nicotinico"
4140 PRINT AT 18,0;"12 Contenido
en Vitamina C"
4170 INPUT "numero: ";num
4180 IF num<=0 OR num>12 THEN GO
 TO 4170
4190 LET l=num
```

```
4200 RETURN
5000 REM datos
5010 REM
5020 RESTORE 5030
5030 DIM l$(12,20)
5040 DATA "agua","proteinas","li
pidos totales","colesterol","hid
ratos de carbono","fibras","Calo
rias","vitamina A","vitamina B1"
,"vitámina B2","acído nicotinico
","vitamina C"
5050 FOR s=1 TO 12: READ l$(s):
NEXT s
5060 RETURN
```

¿Cómo lo harías, Carlos?

CARLOS.—Debe ser bastante difícil; no tengo ni idea de cómo se puede reorganizar una tabla de manera que aparezcan sus componentes ordenados de mayor a menor.

MARGA.—Parece mentira, Carlos, que a los computadores se les llame ordenadores y no sepas utilizar esta característica fundamental.

CARLOS.—Oye, si eres tan lista, ¿por qué no lo haces tú?

FEDE.—Esta vez te has pasado, porque la rutina que es capaz de ordenar los datos no es fácil de hacer.

La rutina es la siguiente:

```
3000 REM ordena
3010 REM
3020 PRINT AT 10,10;"Trabajo": D
IM w(26): DIM z(26)
3030 FOR s=1 TO 26: LET z(s)=n(l
,s)-s*(n(l,s)=0): NEXT s
3040 FOR t=1 TO 26: LET max=z(t)
3050 FOR s=1 TO 26
3070 IF z(s)>=max THEN LET max=z
(s): LET w(t)=s
3080 NEXT s
3090 LET z(w(t))=-999
3100 NEXT t
3110 CLS : PRINT #ca;l;" = ";l$(
l)'': FOR s=1 TO 26
3130 LET y=w(s): PRINT #ca;(" "
AND s<10);s;" ";r$(y);" ";n(l,y)
3140 NEXT s
3150 RETURN
```

¿Podrías explicarla?

MARGA.—Qué pasada, jefe. Creo que otra vez estaré callada.

FEDE.—La rutina empieza en la línea 3020 por escribir en el centro de la pantalla un cartel que le dice al usuario que el computador está trabajando. A continuación, dimensiona dos variables numéricas con un número de valores semejante al de la tabla, en este caso 26. De estas dos variables w contendrá, al final de la ejecución de la rutina, una lista del nuevo

orden que tienen los alimentos. La variable z es una variable temporal para poder ordenarlos. En la línea 3030 empieza un lazo que introduce los correspondientes valores que queremos ordenar de mayor a menor en la variable z. El lazo, cada vez que un alimento tiene como valor 0, sustituye éste por un valor negativo creciente de –1 a –26.

En la línea 3040 comienza otro lazo que, en primer lugar, asigna como valor máximo el valor de z(t), a continuación comienza el siguiente lazo que recorre la tabla de valores de los alimentos contenida en z( ), y si el calor siguiente es mayor o igual que el máximo, asigna a la variable max (valor máximo) este valor y a la variable w(t) el valor del contador de este lazo. Cuando finaliza el lazo interno, la línea 3090 elimina de la tabla z() el valor máximo encontrado, cuyo índice habíamos guardado en w(t), asignándole el número –999, que es el número más negativo de la tabla y termina el lazo externo.

Por tanto, podemos resumir su funcionamiento de la siguiente forma: construye una tabla temporal con el valor que queremos ordenar de los alimentos y otra donde vamos escribiendo los valores máximos de la tabla suprimiendo de la misma el valor máximo encontrado, asignándole el número –999. Al final del lazo la tabla w() contendrá el nuevo orden.

Marga, ¿serías ahora capaz de explicar el resto?

MARGA.—Este programa contiene varias subrutinas, empezaré por la más fácil:

```
4000 REM menu
4010 REM
4020 PRINT AT 0,1;"***CONTENIDO
EN 100 GRAMOS***";AT 2,11;"de FR
UTA"
4030 PRINT AT 7,0;" 1 Contenido
en Agua"
4040 PRINT AT 8,0;" 2 Contenido
en Proteinas"
4050 PRINT AT 9,0;" 3 Contenido
en Lipidos totales"
4060 PRINT AT 10,0;" 4 Contenido
 en Colesterol"
4070 PRINT AT 11,0;" 5 Contenido
en Hidratos Carbono"
4080 PRINT AT 12,0;" 6 Contenido
en Fibras"
4090 PRINT AT 13,0;" 7 Contenido
en Calorias"
4100 PRINT AT 14,0;" 8 Contenido
en Vitamina A"
4110 PRINT AT 15,0;" 9 Contenido
en Vitamina B1"
4120 PRINT AT 16,0;"10 Contenido
en Vitamina B2"
4130 PRINT AT 17,0;"11 Contenido
en Acido Nicotinico"
4140 PRINT AT 18,0;"12 Contenido
en Vitamina C"
4170 INPUT "numero: ";num
4180 IF num<=0 OR num>12 THEN GO
 TO 4170
4190 LET l=num
4200 RETURN
```

Parece que *menú,* líneas 4000 a 4200, lo único que hace es escribir los valores que tenemos guardados en la tabla —líneas 4020 a 4150— para que a continuación podamos escoger el valor que queramos ordenar: calorías, vitaminas, etc.

La línea 4180 evita que pidamos datos que no están en la tabla. Por último, la 4190 asigna a la variable l, que es la que utiliza la *tabla ordena,* el dato que queremos ordenar.

```
5000 REM datos
5010 REM
5020 RESTORE 5030
5030 DIM l$(12,20)
5040 DATA "agua","proteinas","li
pidos totales","colesterol","hid
ratos de carbono","fibras","Calo
rias","vitamina A","vitamina B1"
,"vitamina B2","acido nicotinico
","vitamina C"
5050 FOR s=1 TO 12: READ l$(s):
NEXT s
5060 RETURN
9999 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175:
 PRINT AT 20,1;"numero: █"
```

La rutina siguiente *datos,* no sé muy bien para qué sirve.

CARLOS.—Parece mentira, listilla, creo que sólo sirve para generar una serie de cadenas de caracteres que contengan los nombres de los datos y así se puedan escribir.

FEDE.—Bravo, Carlos, porque la línea 3110 de la rutina *ordena* utiliza esta variable 1$ para que podamos conocer a qué se refieren los datos ordenados.

MARGA.—El resto del programa es muy parecido al anterior, jefe..., podríamos pasar de explicarlo; esto es muy aburrido.

```
2 = proteinas

 1 aguacate        1.7
 2 aceituna        1.5
 3 grosella        1.4
 4 platano         1.3
 5 mora            1.2
 6 higo            1.2
 7 frambuesa       1.1
 8 cereza          1.1
 9 naranja         0.9
10 limon           0.9
11 albaricoque     0.9
12 uva             0.8
13 melocoton       0.8
14 mandarina       0.8
15 fresa           0.8
16 ciruela         0.7
17 toronja         0.6
18 pomelo          0.6
19 melon           0.6
20 arandano        0.6
21 sandia          0.5
22 pera            0.5
23 pina            0.4
24 ananas          0.4
25 membrillo       0.3
26 manzana         0.3
```

8 = vitamina A		
1	melon	3420
2	albaricoque	2790
3	cereza	1000
4	melocoton	880
5	sandia	590
6	platano	430
7	mandarina	420
8	ciruela	350
9	aceituna	300
10	aguacate	290
11	arandano	280
12	mora	200
13	naranja	190
14	frambuesa	150
15	pina	130
16	ananas	130
17	grosella	120
18	manzana	90
19	uva	80
20	higo	75
21	fresa	60
22	pera	20
23	limon	0
24	membrillo	0
25	pomelo	0
26	toronja	0

7 = Calorias		
1	aguacate	245
2	aceituna	132
3	platano	94
4	uva	68
5	frambuesa	66
6	higo	65
7	pera	61
8	arandano	61
9	cereza	60
10	manzana	58
11	grosella	58
12	mora	58
13	membrillo	52
14	albaricoque	51
15	ciruela	50
16	pina	47
17	melocoton	47
18	ananas	47
19	naranja	45
20	mandarina	44
21	toronja	39
22	pomelo	39
23	fresa	37
24	limon	32
25	sandia	28
26	melon	20

12 = vitamina C		
1	fresa	60
2	naranja	49
3	limon	45
4	toronja	40
5	pomelo	40
6	grosella	35
7	melon	33
8	mandarina	31
9	frambuesa	25
10	pina	24
11	ananas	24
12	mora	21
13	manzana	16
14	arandano	16
15	aguacate	16
16	platano	10
17	membrillo	10
18	melocoton	8.7
19	cereza	8
20	albaricoque	8
21	sandia	6
22	ciruela	5
23	uva	4
24	pera	4
25	higo	2
26	aceituna	0

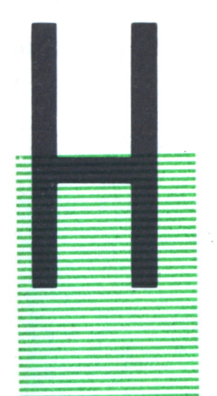
```
***CONTENIDO EN 100 GRAMOS***
          de FRUTA


1  Contenido en Agua
2  Contenido en Proteinas
3  Contenido en Lipidos totales
4  Contenido en Colesterol
5  Contenidoen Hidratos Carbono
6  Contenidoen Fibras
7  Contenidoen Calorias
8  Contenidoen Vitamina A
9  Contenidoen Vitamina B1
10 Contenidoen Vitamina B2
11 Contenidoen Acido Nicotinico
12 Contenidoen Vitamina C

numero:
```

LOS ORDENADORES EN LA ENSEÑANZA DE LAS CIENCIAS NATURALES





Martes, 24 de marzo de 1986

HACE un espléndido día, y para celebrar la llegada de la primavera, Carlos y Marga han decidido ir caminando desde el laboratorio hasta la plaza de Castilla. Marga vive cerca de allí y Carlos tomará después cualquier medio de transporte.

Ambos están entusiasmados con sus avances en informática, y sin darse cuenta han acabado hablando del tema.

CARLOS.—Ayer hablé con Rosa, mi hermana pequeña, y me dijo que han comprado ordenadores en su colegio.

MARGA.—Ya era hora de que la informática llegase a las escuelas, y que no sea sólo una máquina de juegos.

CARLOS.—Tienes razón, puede ser muy útil en el estudio, ya que ofrece posibilidades como el acúmulo de textos, gráficas, relacionar los conocimientos sobre un tema, hacer esquemas bastante precisos e incluso visualizar fenómenos físicos y químicos que de otra forma no sería posible apreciar.

MARGA.—Recuerdo que en el Museo de Ciencias Naturales de Londres hay ordenadores que simulan el montaje de circuitos eléctricos, de péndulos o de otros fenómenos físicos y puedes jugar con ellos de manera que si no lo haces correctamente ocasionas un desastre y te llevas un susto; los padres llevan a sus hijos allí, los niños se divierten, manejan el ordenador y además pueden aprender electricidad: es una idea genial.

CARLOS.—Imagino que los programas serán muy complejos, y los ordenadores muy potentes.

MARGA.—Supongo, pero yo creo que con nuestros conocimientos podríamos hacer algo parecido, aunque más sencillo. Se me ocurre un programa que represente una parte de la anatomía humana, por ejemplo, el esqueleto, los músculos o los órganos de los sentidos, y que tenga los nombres de los componentes desordenados. El alumno habría de señalar en el dibujo cada uno de ellos y el ordenador le indicaría si era correcto o no.

CARLOS.—Claro, emitiría un sonido o encendería un letrero, hay muchos juegos de este tipo.

Además, tiene aplicación en cualquier asignatura y ofrece muchas posibilidades.

MARGA.—¿Manos a la obra?

CARLOS.—¡Manos a la obra!

Ambos entran en una cafetería, piden un refresco y pasan horas hablando en un raro lenguaje... Al final deciden que enseñarán el resultado a Fede.

El jueves siguiente la reunión empieza así:

CARLOS.—Jefe, hemos hecho un programa que dibuja un ojo humano en la pantalla, ¿podríamos comentarlo?

FEDE.—Claro, Carlos, vamos a verlo.

CARLOS.—Este es... dice Carlos, enseñándole a Fede un listado.

```
    5 REM dibuja ojo
    6 REM ___________
   10 RESTORE 1000: FOR s=1 TO 6:
  GO SUB 900: NEXT s
   20 RESTORE 1017: LET a=0: FOR
s=1 TO 2: READ x0,y0,x1,y1: GO S
UB 960: NEXT s
   30 RESTORE 1019: FOR s=1 TO 2:
  GO SUB 900: NEXT s
   40 RESTORE 1021: LET a=0: FOR
s=1 TO 2: READ x0,y0,x1,y1: GO S
UB 960: NEXT s
   50 RESTORE 1023: FOR s=1 TO 4:
  GO SUB 900: NEXT s
   60 RESTORE 1027: LET a=0: FOR
s=1 TO 2: READ x0,y0,x1,y1: GO S
UB 960: NEXT s
   70 RESTORE 1029: FOR s=1 TO 2:
  GO SUB 900: NEXT s
   80 RESTORE 1031: LET a=0: FOR
s=1 TO 14: READ x0,y0,x1,y1: GO
SUB 960: NEXT s
   90 RESTORE 1045: FOR t=1 TO 6:
  READ x0,y0,x1,y1: FOR s=0 TO x1
  STEP 2: PLOT x0,s*y1+y0: NEXT s
: NEXT t
   92 PLOT 174,104: PLOT 174,105:
  FOR s=1 TO 5: PLOT INVERSE 1;17
6,90+s: NEXT s: FOR s=1 TO 5: PL
OT 177,90+s: NEXT s: FOR s=1 TO
5: PLOT 178,90+s: NEXT s
   99 FOR t=2 TO 10: LET pausa=10
0/t: LET numero=t: GO SUB 2000:
NEXT t
  100 LET p$="": LET f$="": LET p
untuacion=0: LET fallos=0: LET n
umero=1: GO SUB 2200
  899 GO TO 9999
  900 REM dibuja curvas ojo
  910 REM ____________________
  930 READ x0,y0,x1,y1,z
  950 LET a=z*PI/180
  960 PLOT x0,y0: DRAW x1-x0,y1-y
0,a
  970 RETURN
 1000 REM Datos del dibujo
 1010 REM _________________
 1011 DATA 84,58,184,102,149
```

```
1012 DATA 80,56,188,100,149
1013 DATA 180,112,84,128,119
1014 DATA 184,116,80,128,119
1015 DATA 84,127,84,59,120
1016 DATA 80,127,80,57,120
1017 DATA 188,101,222,108
1018 DATA 184,115,216,122
1019 DATA 177,111,102,135,91
1020 DATA 180,102,102,47,-130
1021 DATA 177,110,216,118
1022 DATA 181,102,220,110
1023 DATA 174,107,113,138,75
1024 DATA 175,103,110,47,-120
1025 DATA 113,138,105,142,60
1026 DATA 110,47,103,44,-60
1027 DATA 174,106,217,115
1028 DATA 175,104,218,113
1029 DATA 94,72,94,114,70
1030 DATA 94,114,94,72,50
1031 DATA 85,128,85,99
1032 DATA 85,99,87,99
1033 DATA 87,99,90,128
1034 DATA 90,128,92,128
1035 DATA 92,128,92,125
1036 DATA 92,125,96,125
1037 DATA 96,125,101,138
1038 DATA 85,58,85,88
1039 DATA 85,88,87,88
1040 DATA 87,88,90,58
1041 DATA 90,58,92,58
1042 DATA 92,58,92,60
1043 DATA 92,60,96,60
1044 DATA 96,60,102,48
1045 DATA 93,60,12,1
1046 DATA 94,61,12,1
1047 DATA 95,60,12,1
1048 DATA 93,124,12,-1
1049 DATA 94,125,12,-1
1050 DATA 95,124,12,-1
2000 REM Pone nombres
2010 REM ___________________
2020 LET x0=47: LET y0=91: LET x
1=62: LET y1=91: LET c$="Cornea"
: LET n=numero: LET x=0: LET y=1
0: GO SUB 3000
2030 PAUSE pausa
2040 LET x0=80: LET y0=23: LET x
1=80: LET y1=85: LET c$="Camara
anterior": LET n=numero: LET x=0
: LET y=19: GO SUB 3000
2050 PAUSE pausa
2060 LET x0=40: LET y0=123: LET
x1=84: LET y1=123: LET c$="Iris"
: LET n=numero: LET x=1: LET y=6
: GO SUB 3000
2070 PAUSE pausa
2080 LET x0=50: LET y0=107: LET
x1=84: LET y1=95: LET c$="Pupila
": LET n=numero: LET x=0: LET y=
8: GO SUB 3000
2090 PAUSE pausa
2100 LET x0=198: LET y0=60: LET
x1=178: LET y1=92: LET c$="Macul
a": LET n=numero: LET x=25: LET
y=14: GO SUB 3000
2110 PAUSE pausa
2120 RETURN
2200 REM pregunta
2210 REM ______________
2220 LET x0=47: LET y0=91: LET x
1=62: LET y1=91: LET c$="██████"
: LET n=numero: LET x=0: LET y=1
0: GO SUB 3000
2229 LET nuin=0
```

```
2230 INPUT "Nombre=";a$: IF a$<>
"Cornea" THEN LET nuin=nuin+1: B
EEP 0.4,5: LET fallos=fallos+1:
GO SUB 4000: IF nuin<3 THEN GO T
O 2230
2231 IF nuin<3 THEN LET puntuaci
on=puntuacion+3: GO SUB 4000
2240 LET x0=80: LET y0=23: LET x
1=80: LET y1=85: LET c$="[illegible]
[illegible]": LET n=numero: LET x=0
: LET y=19: GO SUB 3000
2249 LET nuin=0
2250 INPUT "Nombre=";a$: IF a$<>
"Camara anterior" THEN LET nuin=
nuin+1: BEEP 0.4,5: LET fallos=f
allos+1: GO SUB 4000: IF nuin<3
THEN GO TO 2250
2251 IF nuin<3 THEN LET puntuaci
on=puntuacion+3: GO SUB 4000
2300 RETURN
3000 REM dibuja rayas
3010 REM
3020 FOR s=1 TO n: PRINT OVER 1;
AT y,x;c$: PLOT OVER 1;x0,y0: DR
AW OVER 1;x1-x0,y1-y0: BEEP 0.02
,10
3050 NEXT s
3060 RETURN
4000 REM representa puntos
4010 REM
4020 LET p$="": FOR y=1 TO puntu
acion
4030 LET p$=p$+"[illegible]"
4040 NEXT y
4050 LET f$="": FOR y=1 TO fallo
s
4060 LET f$=f$+"[illegible]"
4070 NEXT y
4080 PRINT AT 0,0; INK 2;"Aciert
os";p$
4090 PRINT AT 1,0; INK 3;"Fallos
";f$
4100 RETURN
```

FEDE.—Empieza por comentarlo tú y yo después os digo qué me parece.

CARLOS.—El programa consta de siete rutinas bastante independientes. La primera de ellas dibuja el ojo en la pantalla utilizando las otras seis directa o indirectamente.

```
   5 REM dibuja ojo
   6 REM
  10 RESTORE 1000: FOR s=1 TO 6:
 GO SUB 900: NEXT s
  20 RESTORE 1017: LET a=0: FOR
s=1 TO 2: READ x0,y0,x1,y1: GO S
UB 960: NEXT s
  30 RESTORE 1019: FOR s=1 TO 2:
 GO SUB 900: NEXT s
  40 RESTORE 1021: LET a=0: FOR
s=1 TO 2: READ x0,y0,x1,y1: GO S
UB 960: NEXT s
  50 RESTORE 1023: FOR s=1 TO 4:
 GO SUB 900: NEXT s
  60 RESTORE 1027: LET a=0: FOR
s=1 TO 2: READ x0,y0,x1,y1: GO S
UB 960: NEXT s
  70 RESTORE 1029: FOR s=1 TO 2:
 GO SUB 900: NEXT s
```

```
  80 RESTORE 1031: LET a=0: FOR
s=1 TO 14: READ x0,y0,x1,y1: GO
SUB 960: NEXT s
  90 RESTORE 1045: FOR t=1 TO 6:
 READ x0,y0,x1,y1: FOR s=0 TO x1
 STEP 2: PLOT x0,s*y1+y0: NEXT s
: NEXT t
  92 PLOT 174,104: PLOT 174,105:
 FOR s=1 TO 5: PLOT INVERSE 1;17
6,90+s: NEXT s: FOR s=1 TO 5: PL
OT 177,90+s: NEXT s: FOR s=1 TO
5: PLOT 178,90+s: NEXT s
  99 FOR t=2 TO 10: LET pausa=10
0/t: LET numero=t: GO SUB 2000:
NEXT t
 100 LET p$="": LET f$="": LET p
untuacion=0: LET fallos=0: LET n
umero=1: GO SUB 2200
 899 GO TO 9999
 900 REM dibuja curvas ojo
 910 REM ___________________
 930 READ x0,y0,x1,y1,z
 950 LET a=z*PI/180
 960 PLOT x0,y0: DRAW x1-x0,y1-y
0,a
 970 RETURN
1000 REM Datos del dibujo
1010 REM _________________
1011 DATA 84,58,184,102,149
1012 DATA 80,56,188,100,149
1013 DATA 180,112,84,128,119
1014 DATA 184,116,80,128,119
1015 DATA 84,127,84,59,120
1016 DATA 80,127,80,57,120
1017 DATA 188,101,222,108
1018 DATA 184,115,216,122
1019 DATA 177,111,102,135,91
1020 DATA 180,102,102,47,-130
1021 DATA 177,110,216,118
1022 DATA 181,102,220,110
1023 DATA 174,107,113,138,75
1024 DATA 175,103,110,47,-120
1025 DATA 113,138,105,142,60
1026 DATA 110,47,103,44,-60
1027 DATA 174,106,217,115
1028 DATA 175,104,218,113
1029 DATA 94,72,94,114,70
1030 DATA 94,114,94,72,50
1031 DATA 85,128,85,99
1032 DATA 85,99,87,99
1033 DATA 87,99,90,128
1034 DATA 90,128,92,128
1035 DATA 92,128,92,125
1036 DATA 92,125,96,125
1037 DATA 96,125,101,138
1038 DATA 85,58,85,88
1039 DATA 85,88,87,88
1040 DATA 87,88,90,58
1041 DATA 90,58,92,58
1042 DATA 92,58,92,60
1043 DATA 92,60,96,60
1044 DATA 96,60,102,48
1045 DATA 93,60,12,1
1046 DATA 94,61,12,1
1047 DATA 95,60,12,1
1048 DATA 93,124,12,-1
1049 DATA 94,125,12,-1
1050 DATA 95,124,12,-1
```

Así, la rutina *dibuja ojo* comienza en la línea 5 con los habituales REM, para en la línea 10 ejecutar un RESTORE 1000, que permitirá empezar a leer los datos en el orden correcto. El lazo que continúa la línea ejecuta

seis veces la rutina *dibuja curvas del ojo*. Esta rutina, como su nombre indica, dibuja curvas utilizando la instrucción DRAW del Spectrum. Después la comentaremos.

Hasta la línea 92 el programa sigue de la misma manera, es decir, leyendo datos y representando en la pantalla líneas que al final crearán la figura del ojo.

Es interesante destacar que la línea 10 ejecuta *dibuja curvas del ojo* desde su línea 900, mientras que la 20 la ejecuta desde la línea 960. Esto permite dibujar curvas y rectas con la misma rutina y estamos muy satisfechos de haberla programado así, aunque tenemos que confesar que habíamos visto algo parecido en un programa publicado en una revista.

FEDE.—¿En qué revista y quién era el autor del programa? Porque, por otro lado, también se pueden dibujar rectas utilizando la rutina siempre desde la línea 900. En este caso en los DATA de las rectas habría que incluir un cero al final. En realidad, aunque el procedimiento que habéis utilizado es cómodo, las líneas 10 a 90 creo que se pueden resumir en una sola que lea los DATA sucesivamente y los represente, si se incluyen los ceros en los DATA que tienen cuatro datos. Pero, sin duda, el programa es más legible así, aunque bastante más lento.

CARLOS.—No nos acordamos del autor jefe, y lo sentimos porque ya habíamos comentado entre nosotros que a ti no te iba a gustar que no lo dijésemos. Seguimos tu principio de: «Dar al César lo que es del César y a Dios lo que es Dios», y damos las gracias al autor desconocido y a la revista que lo publicó pidiendo perdón por no acordarnos del origen.

Creo que tienes razón con respecto a la primera parte del programa, pero como nos costó bastante trabajo dibujar el ojo, cada curva aceptable la incluimos y se quedó así.

Siguiendo con el programa, la línea 30 dibuja curvas, la 40 rectas, la 50 otra vez curvas y así sucesivamente hasta la 92, que termina el dibujo. El lazo de la línea 99 da valores a las variables número y pausa y ejecuta la rutina *pone nombres*. Por último, la línea 100 da valores a otras variables que necesitará la rutina *pregunta* y la ejecuta.

Como hemos comentado antes, la rutina *dibuja curvas ojo* permite dibujar las curvas del ojo y también las rectas. Empieza en su línea 930 por

```
 100 REM dibuja curvas del ojo
 110 REM ______________________
 120 LET a=z*PI/180
 130 PLOT OVER 1;x0,y0: DRAW OVE
R 1;x1-x0,y1-y0,a
 140 RETURN
```

leer los datos pertinentes con un READ e introducirlos en las variables x0, y0, x1, y1 y z, las variables x0 e y0 indicarán el punto de origen de la curva o de la recta y x1 e y1 el punto final. La instrucción DRAW w,z,m, del Spectrum, dibuja una recta o una curva a partir de un punto en el eje de coor-

denadas previamente definido. La recta tiene una longitud *w* en el eje *x*, *z* en el eje *y*, y es una recta si m = 0 y una curva si *m* es diferente de 0. La rutina aprovecha esta instrucción para dibujar una recta, si *a* es igual a cero, o una curva si *a* es distinto de 0. Hemos hecho que la recta o la curva esté definida en sus extremos, haciendo que dibuje desde x1-x0 a y1-y0. De esta manera, x0 e y0 son los puntos iniciales en los ejes de coordenadas y x1 y y1 los finales, estando la recta definida por sus puntos inicial y final, en vez de punto inicial y longitudes en ambos ejes, que resulta más fácil de utilizar.

El **DRAW** tiene una variable final *m* que sirve para dibujar curvas y nosotros hemos transformado esta variable en *a* tomando el número en el **DATA** y calculando sus correspondientes radianes en la línea 950, al multiplicarlo por **PI** dividido por 180. De esta manera la línea 960 permite, utilizando **PLOT** y **DRAW**, dibujar una recta desde el punto x0,y0 a x1,y1 y una curva entre estos puntos dado un ángulo en grados.

Esta rutina la hemos utilizado exhaustivamente para dibujar el ojo y es muy cómoda.

Para hacer este tipo de dibujo, la primera vez y construir la tabla que pusimos en *datos del dibujo* lo que hicimos fue crear un programita que permite dibujar y borrar una curva en la pantalla, hasta que la que dibuja te gusta y te quedas con ella. El programita es poco más o menos así y fijaros qué bien funciona:

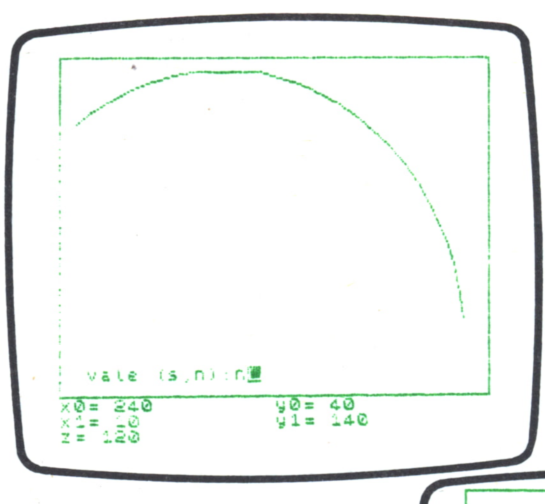
```
  10 INPUT "x0=";x0
  20 INPUT "y0=";y0
  30 INPUT "x1=";x1
  40 INPUT "y1=";y1
  50 INPUT "z=";z
  60 GO SUB 100
  70 INPUT "vale (s,n) ";r$
  80 IF r$<>"s" THEN GO SUB 130:
 GO TO 10
  90 GO TO 9000
 100 REM dibuja curvas del ojo
 110 REM
 120 LET a=z*PI/180
 130 PLOT OVER 1,x0,y0: DRAW OVE
R 1;x1-x0,y1-y0,a
 140 RETURN
9000 REM imprime resultados
9010 REM
9020 LPRINT "x0= ";x0,"y0= ";y0
9030 LPRINT "x1= ";x1,"y1= ";y1
9040 LPRINT "z= ";z
9050 FOR s=1 TO 3: LPRINT : NEXT
 s
9060 GO TO 10
```

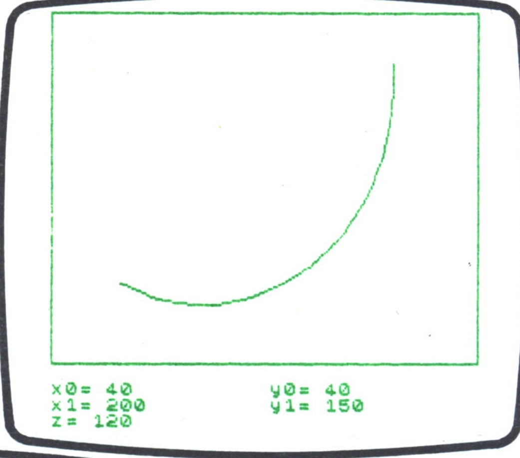
Como veis, el programa te permite dibujar una línea recta o curva y si no te gusta borrarla. Si la línea es la correcta, te imprime los valores que hay que poner en la tabla. De esta manera, y por supuesto gracias a la ayuda de Marga que dibuja bien, hemos dibujado el ojo.

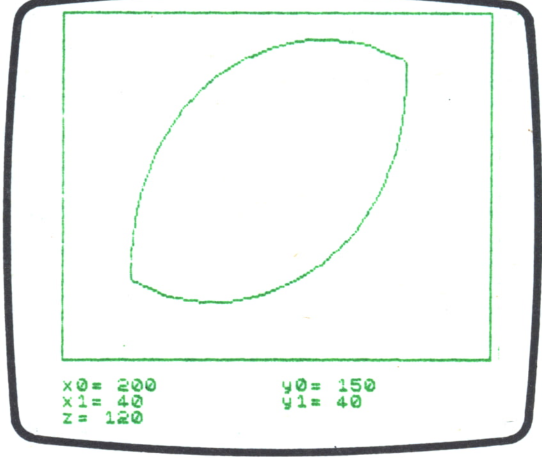
MARGA.—Gracias.

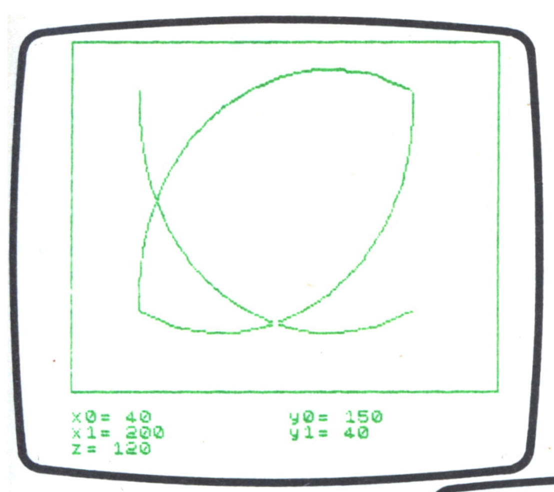
CARLOS.—De nada...

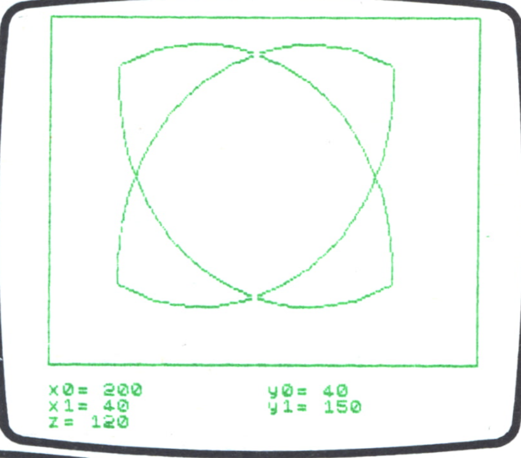
La tabla que va desde la línea 1000 a la 1050 contiene los datos del di-

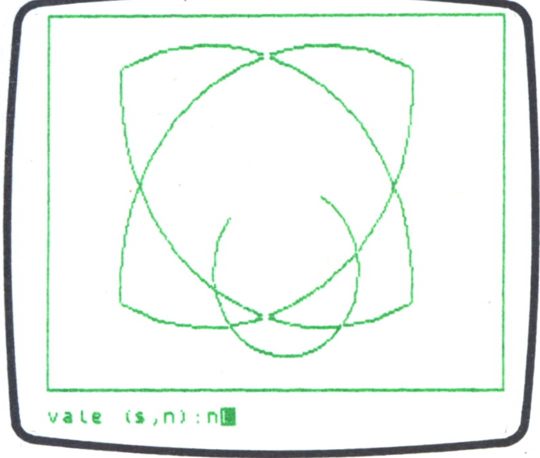

vale (s,n):n
x0= 240
y0= 40
x1= 10
y1= 140
z= 120

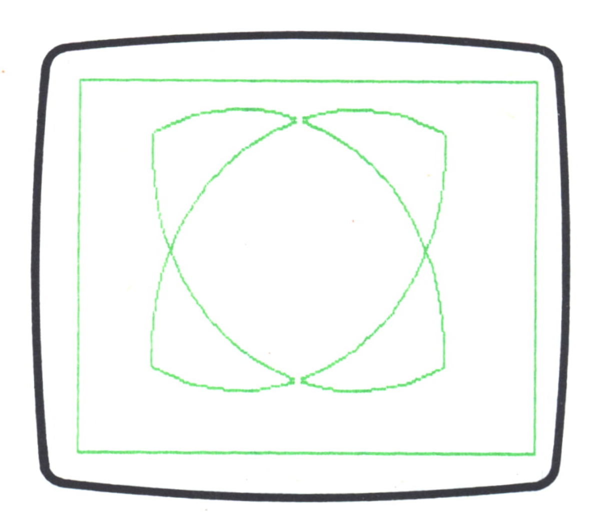

x0= 40
y0= 40
x1= 200
y1= 150
z= 120


x0= 200
y0= 150
x1= 40
y1= 40
z= 120


x0= 40
x1= 200
z= 120
y0= 150
y1= 40


x0= 200
x1= 40
z= 120
y0= 40
y1= 150


vale (s,n):n



bujo. Hay algunos detalles del ojo que se dibujan independientemente de la rutina *dibuja curvas ojo,* pero éstos son muy sencillitos y no creo que merezca la pena explicarlos.

```
2000 REM Pone nombres
2010 REM
2020 LET x0=47: LET y0=91: LET x
1=62: LET y1=91: LET c$="Cornea"
: LET n=numero: LET x=0: LET y=1
0: GO SUB 3000
2030 PAUSE pausa
2040 LET x0=80: LET y0=23: LET x
1=80: LET y1=85: LET c$="Camara
anterior": LET n=numero: LET x=0
: LET y=19: GO SUB 3000
2050 PAUSE pausa
2060 LET x0=40: LET y0=123: LET
x1=84: LET y1=123: LET c$="Iris"
: LET n=numero: LET x=1: LET y=6
: GO SUB 3000
2070 PAUSE pausa
2080 LET x0=50: LET y0=107: LET
x1=84: LET y1=95: LET c$="Pupila
": LET n=numero: LET x=0: LET y=
8: GO SUB 3000
2090 PAUSE pausa
2100 LET x0=198: LET y0=60: LET
x1=178: LET y1=92: LET c$="Macul
a": LET n=numero: LET x=25: LET
y=14: GO SUB 3000
2110 PAUSE pausa
2120 RETURN

3000 REM dibuja rayas
3010 REM
3020 FOR s=1 TO n: PRINT OVER 1;
AT y,x;c$: PLOT OVER 1;x0,y0: DR
AW OVER 1;x1-x0,y1-y0: BEEP 0.02
,10
3050 NEXT s
3060 RETURN
```

La rutina que empieza en la línea 2000 pone los nombres a la figura utilizando, a su vez, a la rutina *dibuja rayas*. Para ello, lo que hace es darle valores a una serie de variables, ejecuta la rutina que dibuja una raya y escribe el nombre en la posición deseada. Las líneas de *pone nombres* son prácticamente iguales dos a dos, de tal manera que cada dos líneas dibuja la raya pertinente y hace una pausa de longitud variable.

FEDE.—¿Por qué no utilizáis la rutina *dibuja curvas del ojo* para dibujar las rayas que identifican los nombres?

CARLOS.—La verdad es que, en nuestro afán de separarlo todo, no lo hemos pensado.

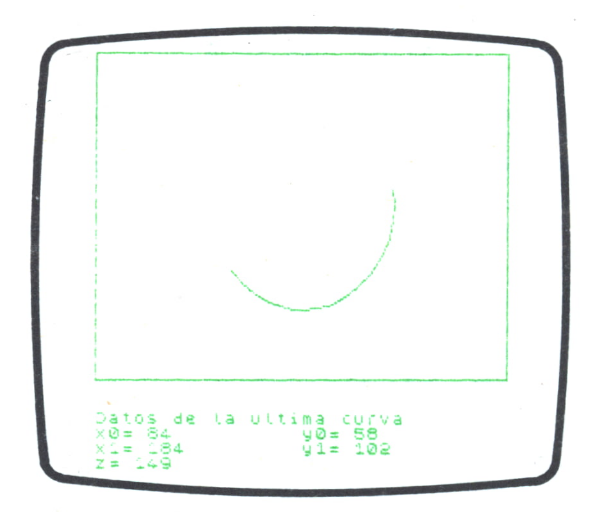
```
2200 REM pregunta
2210 REM
2220 LET x0=47: LET y0=91: LET x
1=62: LET y1=91: LET c$="[illegible]"
: LET n=numero: LET x=0: LET y=1
0: GO SUB 3000
2229 LET nuin=0
2230 INPUT "Nombre=";a$: IF a$<>
"Cornea" THEN LET nuin=nuin+1: B
EEP 0.4,5: LET fallos=fallos+1:
GO SUB 4000: IF nuin<3 THEN GO T
O 2230
2231 IF nuin<3 THEN LET puntuaci
on=puntuacion+3: GO SUB 4000
2240 LET x0=80: LET y0=23: LET x
1=80: LET y1=85: LET c$="[illegible]
[illegible]": LET n=numero: LET x=0
: LET y=19: GO SUB 3000
2249 LET nuin=0
2250 INPUT "Nombre=";a$: IF a$<>
"Camara anterior" THEN LET nuin=
nuin+1: BEEP 0.4,5: LET fallos=f
allos+1: GO SUB 4000: IF nuin<3
THEN GO TO 2250
2251 IF nuin<3 THEN LET puntuaci
on=puntuacion+3: GO SUB 4000
2300 RETURN
```

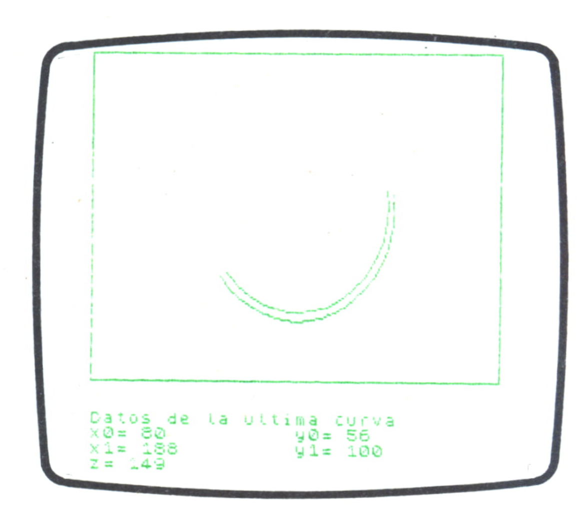
La rutina *pregunta*, que comienza en la línea 2200 y termina en la 2300, utiliza el mismo sistema que la anterior para preguntarle al usuario qué es cada cosa. La única diferencia es que va sumando aciertos y fallos —ver líneas 2229 a 2231— que representará en la pantalla la rutina *representa puntos*. En esta rutina se pueden incluir otros muchos nombres, para hacerlo, lo único necesario es introducir otro bloque de líneas como el que constituyen las 2220 a 2231, o de la 2240 a la 2251.

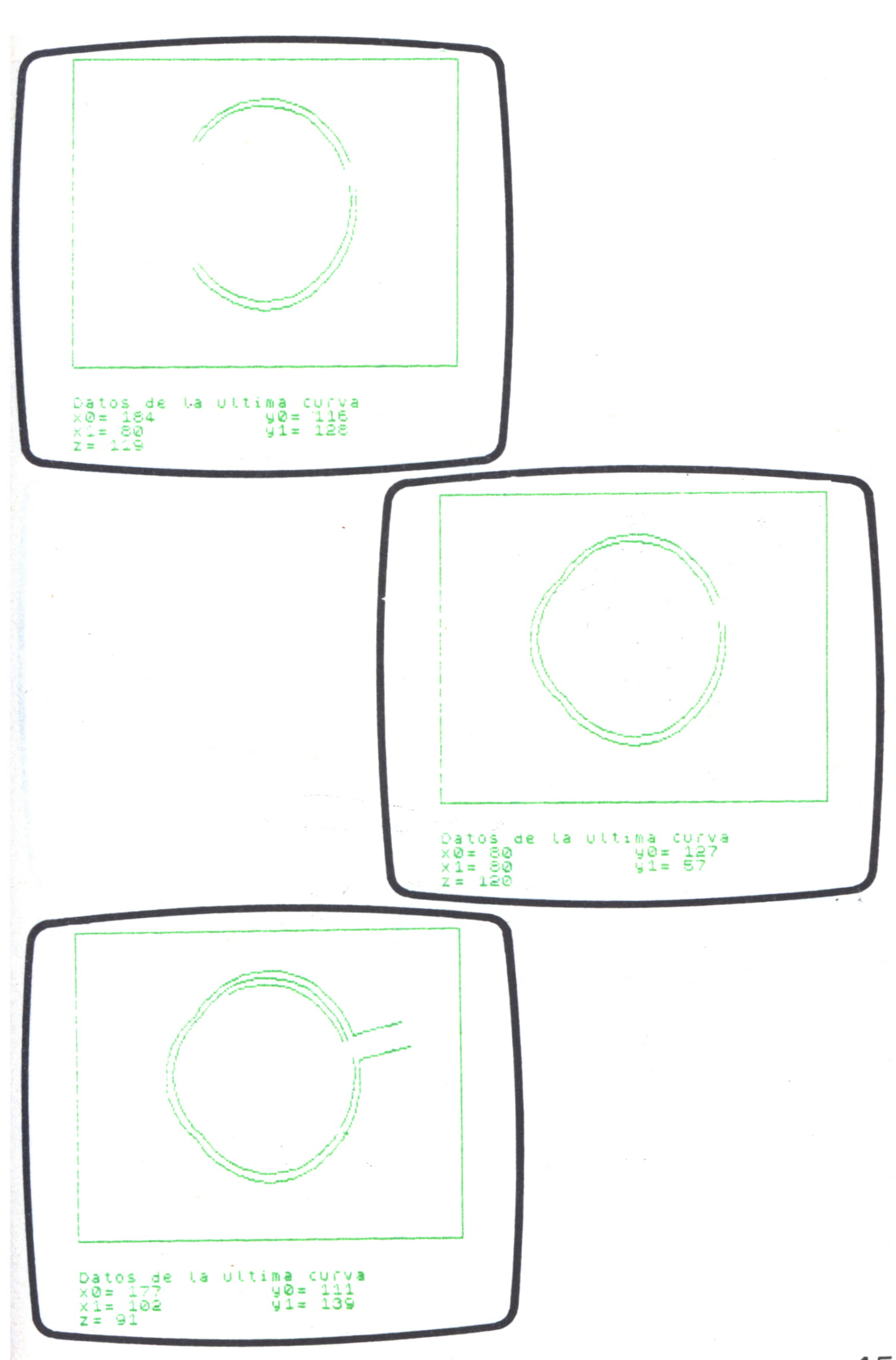
```
4000 REM representa puntos
4010 REM
4020 LET p$="": FOR y=1 TO puntu
acion
4030 LET p$=p$+"[illegible]"
4040 NEXT y
4050 LET f$="": FOR y=1 TO fallo
s
4060 LET f$=f$+"[illegible]"
4070 NEXT y
```

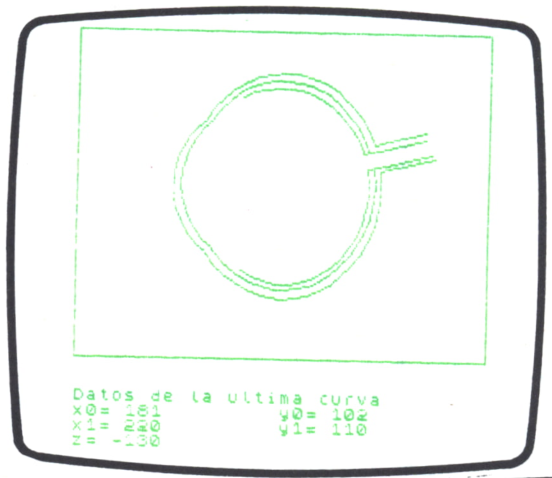
```
4080 PRINT AT 0,0; INK 2;"Aciert
os";p$
4090 PRINT AT 1,0; INK 3;"Fallos
 ";f$
4100 RETURN
```

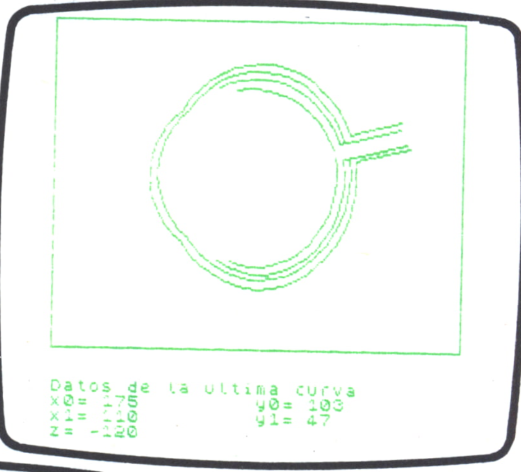
Por último, *representa puntos* pone en la parte superior de la pantalla los aciertos y fallos que el usuario consigue. Para ello inicializa una varible p$ a la que suma una barra tantas veces como aciertos halla, y otra variable f$ hace lo mismo para los fallos. Las líneas 4080 y 4090 escriben las variables p$ y F$ en la pantalla.

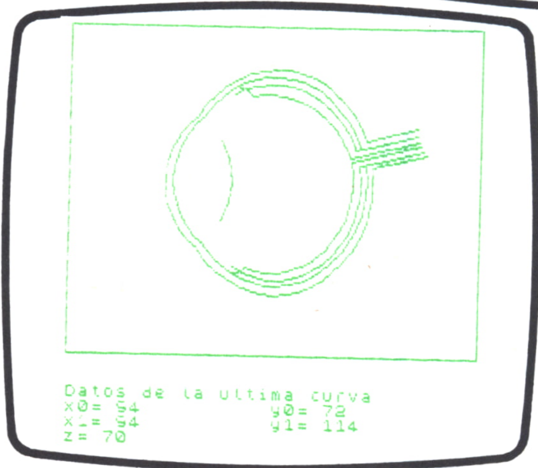


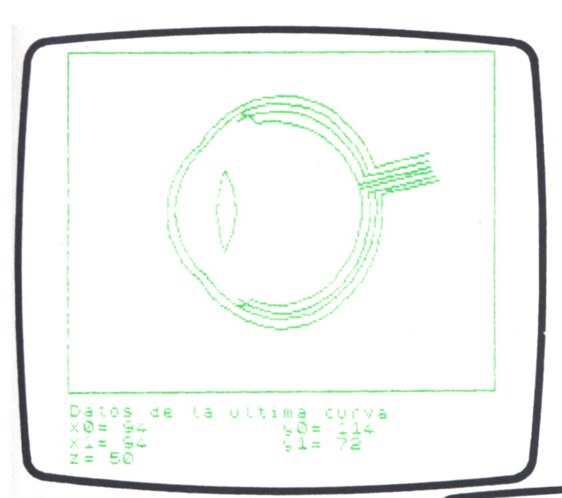


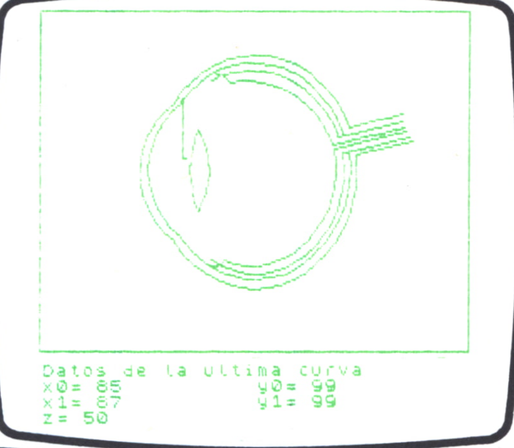

Datos de la ultima curva
x0= 184 y0= 116
x1= 80 y1= 128
z= 119
Datos de la ultima curva
x0= 80 y0= 127
x1= 80 y1= 57
z= 120
Datos de la ultima curva
x0= 177 y0= 111
x1= 102 y1= 139
z= 91

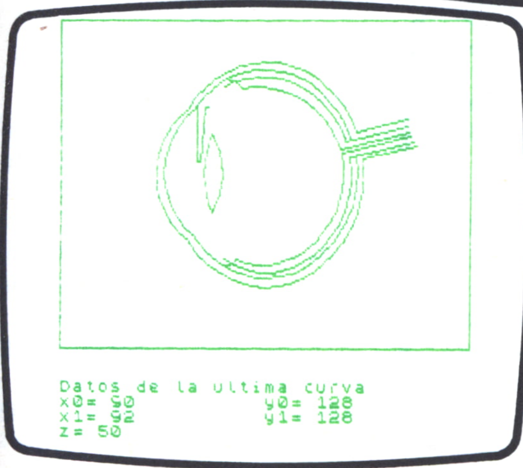

Datos de la ultima curva
x0= 181
y0= 102
x1= 220
y1= 110
z= -130

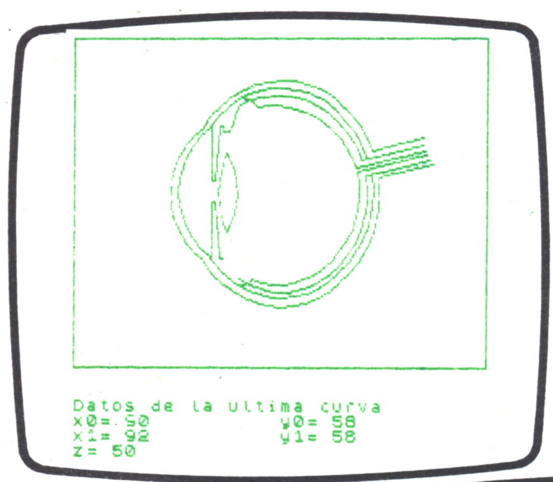

Datos de la ultima curva
x0= 175
y0= 103
x1= 110
y1= 47
z= -120

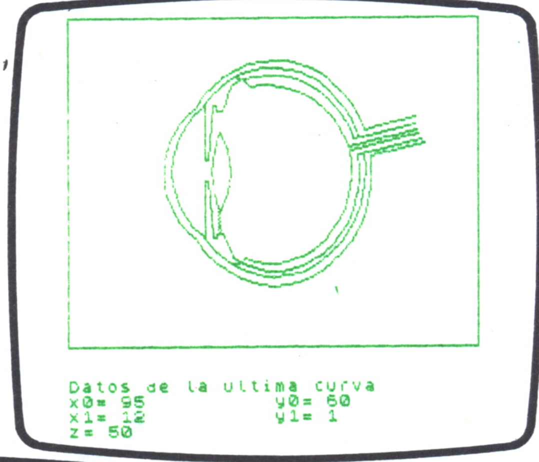

Datos de la ultima curva
x0= 54
y0= 72
x1= 94
y1= 114
z= 70

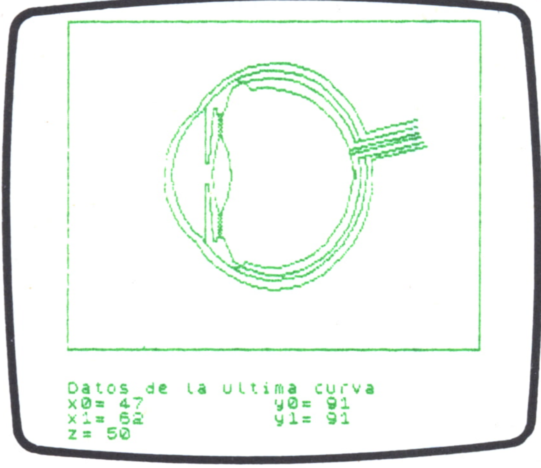

Datos de la ultima curva
x0= 94 y0= 114
x1= 94 y1= 72
z= 50

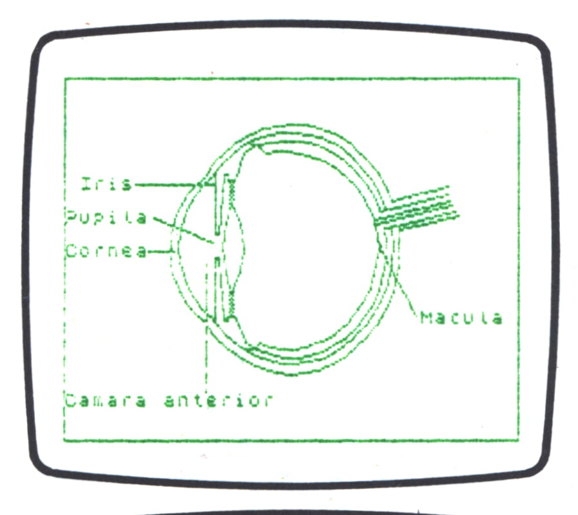

Datos de la ultima curva
x0= 85 y0= 99
x1= 87 y1= 99
z= 50

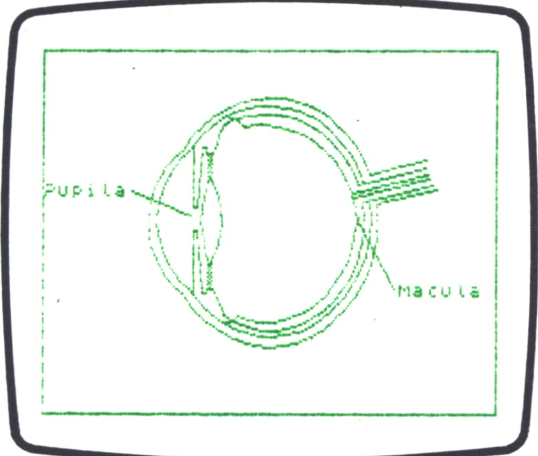

Datos de la ultima curva
x0= 90 y0= 128
x1= 92 y1= 128
z= 50


Datos de la ultima curva
x0= 50 y0= 58
x1= 92 y1= 58
z= 50


Datos de la ultima curva
x0= 95 y0= 60
x1= 12 y1= 1
z= 50


Datos de la ultima curva
x0= 47 y0= 91
x1= 62 y1= 91
z= 50



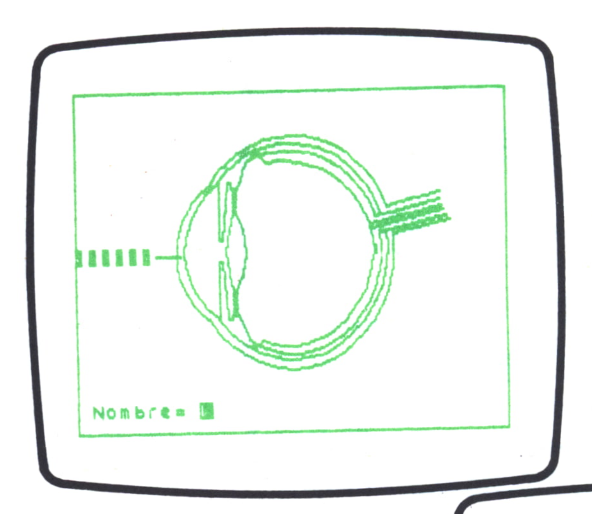


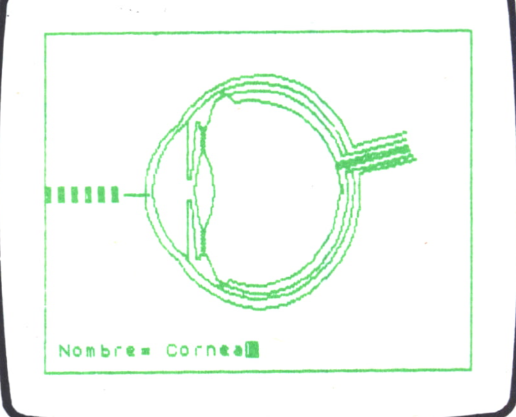
CARLOS.—Marga, ¿tú crees que programas como éste llegarán a venderse en el Rastro, y que los chicos dedicarán la mañana del domingo a buscarlos?

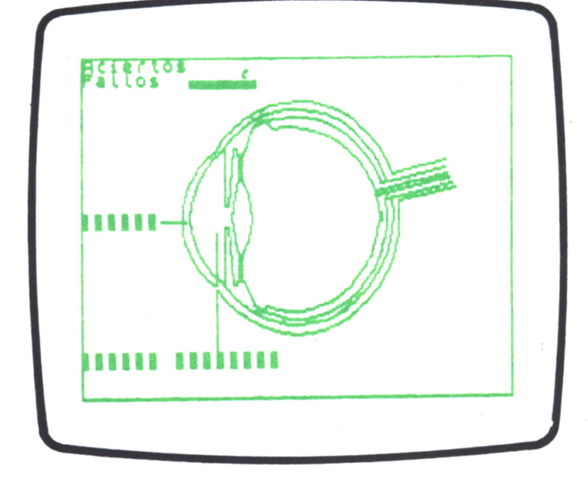
MARGA.—Espero, por el bien de los programadores, que no se vendan en el Rastro ningún tipo de programa, y que los chicos los compren en tiendas especializadas como Dios manda. ¡Sabrás que la mayoría de estas cintas son piratas!

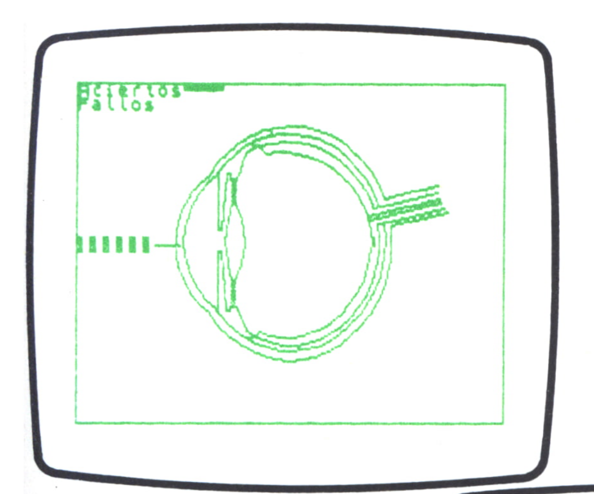
FEDE.—Tú siempre tan reivindicativa. Servirías para defensora... de los negociantes.

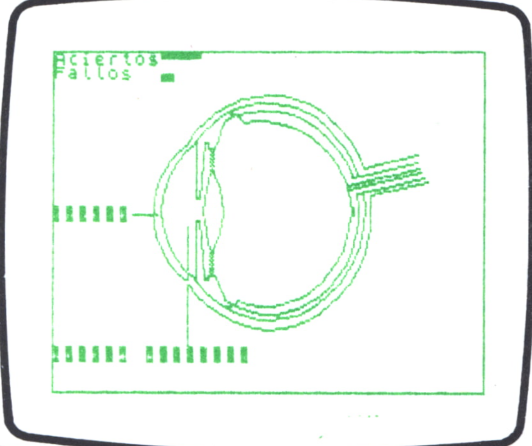
Una carcajada general pone fin al seminario de hoy. Marga y Carlos siguen trabajando; Fede se dispone a abandonar el laboratorio y en la pantalla del ordenador hay un ojo dibujado con flechas por todos lados.

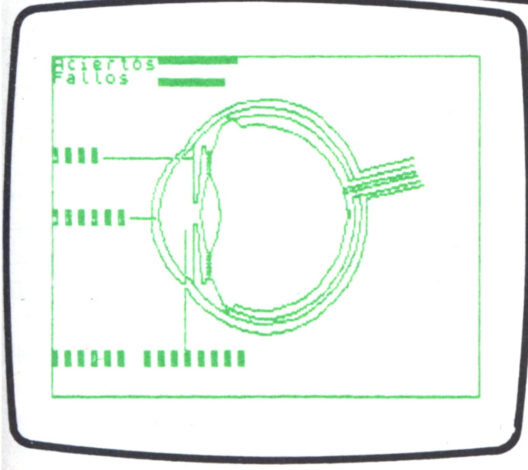

Nombre=


Nombre= Cornea

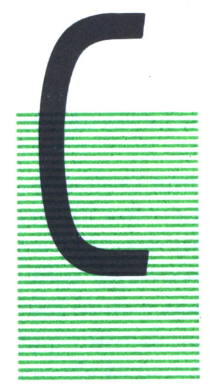

Aciertos
Fallos


Aciertos
Fallos


Aciertos
Fallos


Aciertos
Fallos

CLAVE DICOTOMICA 3



Lunes, 8 de abril de 1986

CARLOS ha terminado pronto de hacer los experimentos propuestos para el día de hoy; tras anotar los resultados en su cuaderno de protocolos y ordenar el laboratorio, ha empezado a hojear una guía de plantas que Fede tenía entre los muchos libros amontonados en su estantería. Como todos sabemos, él es biólogo y, aunque se ha dedicado a la rama de la bioquímica, nunca ha querido dejar olvidadas las otras especialidades de su carrera. Es una cuestión sentimental...

Marga sigue dando vueltas por el laboratorio, dispuesta a no terminar hoy a las 9 de la noche, como últimamente ocurre siempre. No sabe cómo lo hace, pero llega a casa tan cansada que no tiene ganas de ponerse a estudiar; ha decidido distribuir mejor el trabajo y dejar más tiempo para sus estudios, ya que si no, no ve muy claro que pueda acabar la carrera. Carlos y Fede piensan que no tiene razón, que está capacitada para hacer ambas cosas y que todo es cuestión de organizarse un poco, pero ella no llega a creerlos.

Marga se acerca a Carlos, le mira de reojo y decide preguntarle...

MARGA.—¿Qué haces?

CARLOS.—Estaba viendo esta guía de Gimnospermas.

MARGA.—Eso son plantas, ¿no?

CARLOS.—Sí, siempre me han interesado las ramas de la biología que se dedican a describir y clasificar a los seres vivos, como la botánica, la microbiología, la antropología o la zoología. De ellas aprendes a conocer a los organismos que te rodean, su diversidad, sus peculiaridades, sus modos de vida, cómo se relacionan...

MARGA.—A mí nunca me interesaron mucho ese tipo de cosas, soy animal de laboratorio.

CARLOS.—Eso es muy fácil de decir, pero las proteínas, los ácidos nucleicos y las demás moléculas biológicas no andan desperdigadas por el mundo o inmersas en la solución de una probeta, forman parte de estruc-

turas más complejas que realizan funciones que las moléculas por sí solas no podrían hacer.

MARGA.—Ya lo sé, pero tampoco es necesario estudiarse esas horribles clasificaciones que nadie es capaz de entender y que se olvidan a los dos días.

CARLOS.—Las ciencias de que te he hablado no sólo se dedican a la sistemática como tal, sino que sus intereses son otros. Conociendo las características de una especie puedes saber en qué condiciones vive o ha vivido en una época no muy lejana, cuáles han sido sus pautas evolutivas y qué lugar ocupa dentro de la escala filogenética. No es que Linneo tuviera ganas de liarlo todo o de fastidiar a los estudiantes de biología. Además, hacen que te des cuenta de que todos los seres vivos tenemos una composición muy similar y que no es algo anecdótico ser como somos.

MARGA.—Ahora que lo dices, nunca había pensado que pudieran saberse tantas cosas por una simple clasificación...

CARLOS.—No creo que los que se dedican a hacerlas opinen lo mismo. Cuando aparecen recogidas de forma tan clara y ordenada en los libros, parece que no suponen trabajo, pero son muchas las cosas que hay que tener en cuenta a la hora de realizarlas y, sobre todo, si quieres llegar a determinar con toda certeza de qué especie se trata. Las pruebas que se realizan no son sólo de tipo morfológico, ni siquiera microscópicas, sino que ha de recurrirse a pruebas bioquímicas y genéticas en la mayoría de las ocasiones, sobre todo, en el caso de los microorganismos, que tienen una gran capacidad de cambio.

MARGA.—No sabía que fuera tan difícil hacer una clave. Aunque ahora pienso que incluso en nuestro laboratorio es necesaria, ya que trabajamos con material animal y vegetal del que muchas veces no sabemos cuál es su procedencia.

CARLOS.—En cualquier laboratorio o lugar de trabajo es útil, lo que ocurre es que, en nuestro afán de especialización, nos olvidamos de que las moléculas con las que trabajamos proviene de un organismo vivo y de cuáles son las características de dicho organismo. Tenemos una visión demasiado química del asunto y perdemos el punto de referencia de la biología.

MARGA.—¿Me acusas de que no soy capaz de entender cómo ocurren las cosas dentro de los organismos vivos?

CARLOS.—No, sólo digo que nuestra formación es muy parcial, que nunca hemos de olvidar que en el mundo de la biología, como en cualquier otra ciencia, todo guarda una estrecha relación.

MARGA.—Pero no podemos intentar acaparar todos los campos, porque no tendríamos tiempo aunque viviéramos mil años.

CARLOS.—Ya lo sé, pero podemos intentar no perder de vista aquellas partes de la ciencia que nos puedan servir en nuestro trabajo. Este es un caso.

MARGA.—¿Y cómo se estructuran estas claves?

CARLOS.—Normalmente se plantean como *claves dicotómicas* en las que, entre dos o más opciones, has de elegir una. La elección primera te permite acceder a una segunda dicotomía, en la que volverás a elegir; así consigues llegar desde la división, el orden, la familia, hasta la especie o incluso la subespecie o variedad de que se trate, y de paso conoces características peculiares de la planta o animal que estás determinando y en las que no te habrías fijado si no hubiera sido por la clasificación.

MARGA.—Parece divertido...

CARLOS.—Lo es, aprendes muchas cosas sobre el hábitat de la especie, sobre sus características estructurales y funcionales y, una vez determinada, eres capaz de reconocer una especie similar, aunque a simple vista parezca muy diferente.

MARGA.—¿Qué te parece si intentamos hacer un programa con este fin? Sería muy útil tener incluidas en el ordenador las características de las plantas de que proceden nuestras proteínas y comprobar si el material nuevo es o no aquél que creemos.

CARLOS.—Es buena idea, así no tendremos problemas, ya que, aunque hay especies muy similares, puede que sus características bioquímicas sean distintas, y nos volveríamos locos cuando los resultados no fueran los que esperamos.

MARGA.—Es fundamental saber con qué se trabaja. Además, aunque no soy biólogo, puede aprender cómo funcionan las claves. Siempre acabas convenciéndome, no sé cómo lo haces...

CARLOS.—Me alegro, bienvenida al club de los enamorados de las colecciones de plantas y animales.

Carlos y Marga se han tomado muy en serio el problema de las claves dicotómicas; durante una semana han estado estudiando cuál es la mejor forma de hacer un programa de este tipo en el que cada vez que el usuario toma una determinación, el ordenador debe seleccionar entre la información que tiene y plantear nuevas alternativas. Al final debe mostrar las características de cada especie...

FEDE.—Me parece muy interesante el tema que habéis elegido para el seminario de hoy. Creo que los ordenadores pueden ser muy útiles en este tipo de determinaciones.

MARGA.—Además, se aprenden muchas cosas haciendo programas como éste. En esta semana he aprendido más sobre los seres vivos que en toda mi vida.

CARLOS.—Ya te lo dije.

MARGA.—Como siempre, creo que es conveniente hacer un poco de historia antes de empezar...

El interés del hombre por clasificar a los organismos que le rodeaban viene de lejos. El primer intento de clasificación serio que se hizo se atribuye a Aristóteles, el gran filósofo y biólogo griego. El concepto de espe-

cie fue acuñado por el inglés John Ray en el siglo XVII, pero el florecimiento de la sistemática vendría un siglo después con la obra de Carlos Linneo, naturalista sueco, que trabajó en la Universidad de Uppsala.

El esquema de esta clasificación se recoge en su obra *Sistema Natural,* y se caracteriza por seguir un sistema jerárquico creciente.

CARLOS.—Así es, aparecen los llamados taxones, que son: especie, género, familia, orden, clase...

FEDE.—Es curiosa la forma de denominar a las especies, utilizando algo similar a nuestro nombre y apellido.

MARGA.—También esta nomenclatura se debe a Linneo, se conoce como *nomenclatura binomial,* y es un nombre latino o latinizado compuesto por dos palabras:

Primera: nombre del género, escrito con la primera letra mayúscula.

Segunda: denominación de especie, que se escribe con minúsculas y que indica una característica o adjetivo de la misma.

Si la especie se divide a su vez en subespecies, aparece una *nomenclatura trinomial.*

Pongamos un ejemplo:

Acetobacter aceti

es la bacteria del vinagre; su nombre se compone de dos palabras que indican:

Acetobacter, referente al género, es un sustantivo. *Aceti,* referente a la especie, es un adjetivo que indica su capacidad de producir ácido acético.

Dentro de la misma especie se pueden diferenciar varias subespecies:

Acetobacter aceti subsp. *aceti* Pasteur.
Acetobacter aceti subsp. *orleanensis* Hennenberg.
Acetobacter aceti subsp. *xylinum* De Lay and Frateur.

En este caso, la nomenclatura es trinomial.

Detrás de cada nombre aparece quién fue el primero en describir dicha especie.

FEDE.—Esto me recuerda a mis tiempos de estudiante, con la gran cantidad de nombres que tuve que aprender...

Bien, vamos con el programa que habéis hecho.

CARLOS.—Antes que nada, he de decir que aunque los programas de este tipo parecen muy sencillos, la realidad es otra y plantean muchas dificultades, sobre todo a la hora de ordenar los datos introducidos. Estos programas necesitan utilizar un sistema para guardar información en soporte magnético, debido a que manejan grandes cantidades de información que no es fácil guardar en la memoria RAM del computador y alarga innecesariamente los programas.

Hemos construido el esqueleto de este programa guardando la infor-

mación con instrucciones **DATA** de **BASIC**, pero obviamente no es el sistema idóneo para esta función, ni tampoco los microdrives de **SINCLAIR**. Lo hemos hecho así por varios motivos:

Primero, de esta manera se pueden ver los datos en el listado del programa, lo que hace más fácil su comprensión.

Segundo, los microdrives de **SINCLAIR** son más útiles para guardar programas que para buscar información en un archivo.

Tercero, lo importante es el mecanismo que permite guardar y manipular este tipo de información.

Aunque el programa que hemos propuesto tampoco es el idóneo en este tipo de trabajo, sí se ha procurado que sea lo más fácil posible en cuanto a la introducción de datos y la manipulación de los mismos, cosa muy importante en los ordenadores personales con poca capacidad de memoria.

Empecemos con el programa:

```
  10 LET nd=39: GO SUB 9900
  20 GO SUB 9000
 100 REM todos los datos
 110 REM ____________________
 120 LET dato=1: GO SUB 9300: GO
 SUB 9200
 130 PAUSE 0: CLS
 140 LET dato=2: GO SUB 9300: GO
 SUB 9200
 150 PAUSE 0: CLS
 160 LET dato=3: GO SUB 9300: GO
 SUB 9200
 161 PAUSE 0: CLS
 162 LET dato=4: GO SUB 9300: GO
 SUB 9200
 170 PAUSE 0: CLS
 200 REM clave dicotomica
 210 REM ____________________
 220 LET y$="": CLS : PRINT "1 "
;a$(1)'"2 ";a$(2)'"3 No sabe, no
 contesta": INPUT "¿ Cual ? (1,2
,3) :";r
 230 IF r<>1 AND r<>2 AND r<>3 T
HEN GO TO 220
 240 LET y$=("0" AND r=3)+("1" A
ND r=1)+("2" AND r=2)+y$
 250 CLS : PRINT "1 ";a$(4)'"2 "
;a$(3)'"3 No sabe, no contesta":
 INPUT "¿ Cual ? (1,2,3) :";r
 260 IF r<>1 AND r<>2 AND r<>3 T
HEN GO TO 220
 270 LET y$=("0" AND r=3)+("1" A
ND r=1)+("2" AND r=2)+y$
 998 IF y$(1)="1" AND y$(2)="1"
THEN CLS : PRINT AT 10,6;"Es una
 Angiosperma"
 999 GO TO 9999
1000 REM CLAVE
1010 REM _____
1012 DATA 1,"Plantas con semilla
s"
1014 DATA 2,"Plantas sin semilla
s"
1016 DATA 3,"Plantas sin flores"
1018 DATA 4,"Plantas con flores"
```

```
1020 DATA 5,"Plantas con esporan
gios,a veces agrupados en estrób
ilos"
1022 DATA 6,"Plantas con semilla
s contenidas en frutos o en estr
óbilos"
1024 DATA 7,"Plantas con conos m
asculinos y  femeninos"_
1026 DATA 8,"Plantas leñosas, a
menudo resi- nosas"
1028 DATA 9,"Plantas con rudimen
tos seminalesencerrados en un ov
ario"
1030 DATA 10,"Plantas con estamb
res y/o pisti-lo"
1032 DATA 11,"Plantas con rudime
ntos seminalesal descubierto"
1034 DATA 12,"del griego pterido
s: helecho y  fitos:planta"
1036 DATA 13,"Plantas herbáceas
o leñosas, ge-neralmente no resi
nosas"
1038 DATA 14,"del griego; gimno=
desnudo y      sperma=semilla"
1040 DATA 15,"del griego; angios
=vaso y        sperma=semilla"
1042 DATA 16,"Plantas con tallos
 articulados"
1044 DATA 17,"Plantas con tallos
 no articulados"
1046 DATA 18,"Plantas con hojas
en forma de    escama"
1048 DATA 19,"Plantas generalmen
te parduscas"
1050 DATA 20,"Plantas con fructi
ficaciones     carnosas"
1052 DATA 21,"Plantas arbustivas
"
1054 DATA 22,"Plantas con hojas
verticiladas"
1056 DATA 23,"Plantas con hojas
verdes"
1058 DATA 24,"Plantas con hojas
aciculares"
1060 DATA 25,"Plantas con fructi
ficaciones se-cas o carnosas"
1062 DATA 26,"árboes, raramente
ar- bustos"
1064 DATA 27,"Plantas con hojas
opuestas o     verticiladas"
1066 DATA 28,"Plantas con estrób
ilos que tienen brácteas opuesta
s o verticila-das"
1068 DATA 29,"Plantas con hojas
alternas"
1070 DATA 30,"Plantas con estrób
ilos que tienen brácteas en espi
ral"
1072 DATA 31,"Plantas con semill
as solitarias"
1074 DATA 32,"Plantas dioicas"
1076 DATA 33,"Plantas con muchas
 semillas"
1078 DATA 34,"Plantas con estrób
ilos"
1080 DATA 35,"Plantas con semill
as no carnosas"
1082 DATA 36,"Plantas monoicas"
1084 DATA 37,"Plantas resinosas"
1086 DATA 38,"Ephedraceae"
1088 DATA 39,"del griego; éphedr
a= asentada    sobre"
8000 REM orden
8010 REM [redacted]
```

```
8020 DATA 4,"12,2,3,5","Pteridof
itas",6,"14,1,6,7,8,11","Gimnosp
ermas",6,"15,1,4,6,10,13","Angio
spermas"
8022 DATA 6,"39,16,18,19,20,21",
"Ephedraceae"
9000 REM carga
9010 REM
9020 RESTORE 1000: DIM a$(nd,96)
: DIM n(nd)
9030 FOR s=1 TO nd
9040 READ n(s),a$(s)
9050 NEXT s
9070 RETURN
9200 REM escribe
9210 REM
9240 FOR s=1 TO planta
9250 FOR t=96 TO 1 STEP -1: IF a
$(p(s),t)<>" " THEN GO TO 9260
9254 NEXT t
9260 PRINT INK 2 AND s=1;a$(p(s)
,1 TO t)
9270 NEXT s
9280 RETURN
9300 REM lee orden
9310 REM
9320 DIM w$(1,32): DIM r$(1,96):
 DIM p(30): LET con=0: RESTORE 8
000
9330 FOR s=1 TO dato
9340 READ planta,r$(1),w$(1)
9350 NEXT s: LET s=s-1
9360 LET f$="": FOR t=1 TO 96: I
F r$(1,t)=" " THEN GO TO 9380
9370 NEXT t
9390 FOR u=1 TO t: IF r$(1,u)=",
" THEN LET con=con+1: LET p(con)
=VAL f$: LET f$="": GO TO 9400
9392 LET f$=f$+r$(1,u)
9400 NEXT u
9410 LET con=con+1: LET p(con)=V
AL f$
9420 PRINT BRIGHT 1;w$(1)''
9430 RETURN
9900 REM caracteres
9910 REM
9920 DATA 4,8,56,4,60,68,60,0
9930 DATA 4,8,56,68,120,64,62,0
9940 DATA 4,8,0,24,8,8,28,0
9950 DATA 4,8,56,68,68,68,56,0
9960 DATA 4,8,68,68,68,68,56,0
9970 DATA 124,0,120,68,68,68,68,
0
9990 DATA 126,0,98,98,82,74,70,0
9991 DATA 0,16,0,16,16,16,16,0
9992 DATA 0,16,0,16,32,66,60,0
9995 RESTORE 9900: FOR s=USR "a"
 TO USR "i"+7: READ c: POKE s,c:
 NEXT s: RETURN
```

Como siempre, es un programa flexible que permite obtener la información ordenada de distintas maneras.

```
1000 REM CLAVE
1010 REM
1012 DATA 1,"Plantas con semilla
s"
```

```
1014 DATA 2,"Plantas sin semilla
s"
1016 DATA 3,"Plantas sin flores"
1018 DATA 4,"Plantas con flores"
1020 DATA 5,"Plantas con esporan
gios,a veces agrupados en estrób
ilos"
1022 DATA 6,"Plantas con semilla
s contenidas en frutos o en estr
óbilos"
1024 DATA 7,"Plantas con conos m
asculinos y  femeninos"
1026 DATA 8,"Plantas leñosas, a
menudo resi- nosas"
1028 DATA 9,"Plantas con rudimen
tos seminalesencerrados en un ov
ario"
1030 DATA 10,"Plantas con estamb
res y/o pisti-lo"
1032 DATA 11,"Plantas con rudime
ntos seminalesal descubierto"
1034 DATA 12,"del griego pterido
s: helecho y  fitos:planta"
1036 DATA 13,"Plantas herbáceas
o leñosas, gé-neralmente no resi
nosas"
1038 DATA 14,"del griego; gimno=
desnudo y       sperma=semilla"
1040 DATA 15,"del griego; angios
=vaso y         sperma=semilla"
1042 DATA 16,"Plantas con tallos
 articulados"
1044 DATA 17,"Plantas con tallos
 no articulados"
1046 DATA 18,"Plantas con hojas
en forma de    escama"
1048 DATA 19,"Plantas generalmen
te parduscas"
1050 DATA 20,"Plantas con fructi
ficaciones      carnosas"
1052 DATA 21,"Plantas arbustivas
"

1054 DATA 22,"Plantas con hojas
verticiladas"
1056 DATA 23,"Plantas con hojas
verdes"
1058 DATA 24,"Plantas con hojas
aciculares"
1060 DATA 25,"Plantas con fructi
ficaciones se-ças o carnosas"
1062 DATA 26,"árboes, raramente
ár- bustos"
1064 DATA 27,"Plantas con hojas
opuestas o      verticiladas"
1066 DATA 28,"Plantas con estrób
ilos que tienen brácteas opuesta
s o verticila-das"
1068 DATA 29,"Plantas con hojas
alternas"
1070 DATA 30,"Plantas con estrób
ilos que tienen brácteas en espi
ral"
1072 DATA 31,"Plantas con semill
as solitarias"
1074 DATA 32,"Plantas dioicas"
1076 DATA 33,"Plantas con muchas
 semillas"
1078 DATA 34,"Plantas con estrób
ilos"
1080 DATA 35,"Plantas con semill
as no carnosas"
1082 DATA 36,"Plantas monoicas"
1084 DATA 37,"Plantas resinosas"
1086 DATA 38,"Ephedraceae"
1088 DATA 39,"del griego; éphedr
a= asentada   sobre"
```

Los datos están en dos tablas: en la primera rutina *clave*, tiene los DATA dispuestos por orden de entrada y permite introducir nuevos datos siguiendo el orden correlativo establecido. Por ejemplo:

1042 DATA 16, "xxx"

En una segunda rutina, *orden*, hemos introducido tres datos nuevos con respecto a los de la anterior.

```
8000 REM orden
8010 REM
8020 DATA 4,"12,2,3,5","Pteridof
itas",6,"14,1,6,7,8,11","Gimnosp
ermas",6,"15,1,4,6,10,13","Angio
spermas"
8022 DATA 6,"39,16,18,19,20,21",
"Ephedraceae"
```

Para cada división: Pteridofitas, Magnoliophyta (Angiospermas), etc.:

1. El número de datos de la tabla que se refieren a la división.
2. El orden de los datos.
3. El nombre de la división.

La parte más importante y que supone más trabajo son los datos del orden, por ejemplo; las pteridofitas:

Según el DATA 12: procede del griego; *pteridos:* helecho y *phitos:* planta.

Según el DATA 2: son plantas sin semillas.

Según el DATA 3: son plantas sin flores.

Según el DATA 5: son plantas con esporangios agrupados a veces en estrobilos.

El orden de los DATA está introducido en la cadena de caracteres:

"12,2,3,5"

Por último, la tabla *orden* lleva a continuación el nombre de la división de que se trata, en este caso, «Pteridofitas».

Una vez que sabemos cómo se entran y están ordenados los datos, explicaremos el funcionamiento del programa.

```
9000 REM carga
9010 REM
9020 RESTORE 1000: DIM a$(nd,96)
: DIM n(nd)
9030 FOR s=1 TO nd
9040 READ n(s),a$(s)
9050 NEXT s
9070 RETURN
```

La línea 10 asigna a la variable nd el número de datos y ejecuta la rutina *caracteres,* que permite escribir con acentos y eñes.

La línea 20 ejecuta la rutina *carga*, que carga todos los datos de clave en dos variables a$( ) y n( ). De estas dos variables n( ) contendrá el número ordinal de entrada en la tabla, lo que nos permite buscar un dato sin necesidad de recorrer la tabla entera; a$( ) contendrá las cadenas de caracteres de los DATA de *clave*. En BASIC, estas cadenas de caracteres tienen que ser, obligatoriamente, de longitud fija; nosotros las hemos fijado en 96 caracteres. El BASIC de SINCLAIR, cuando dimensiona una variable de este tipo, la rellena inicialmente de 96 blancos, pero esto no ocurre en todos los BASIC, ya que algunos no crean la variable hasta que no se le da un valor y, en este caso, es importante poner blancos desde el final de la variable hasta los 96 caracteres, ya que si no, no se puede escribir.

El lazo de las líneas 9030 a 9050 lee los DATA de clave y los introduce en las correspondientes variables a$( ) y n( ). En realidad, nos hemos pasado, ya que n( ) no es necesaria, porque nuestros datos están ordenados sucesivamente, con lo cual cualquier lazo podría encontrarlos. Sin embargo, la existencia de n( ) hace el programa más comprensible.

```
9300 REM lee orden
9310 REM
9320 DIM w$(1,32): DIM r$(1,96):
 DIM p(30): LET con=0: RESTORE 8
000
9330 FOR s=1 TO dato
9340 READ planta,r$(1),w$(1)
9350 NEXT s: LET s=s-1
9360 LET f$="": FOR t=1 TO 96: I
F r$(1,t)=" " THEN GO TO 9380
9370 NEXT t
9390 FOR u=1 TO t: IF r$(1,u)=",
" THEN LET con=con+1: LET p(con)
=VAL f$: LET f$="": GO TO 9400
9392 LET f$=f$+r$(1,u)
9400 NEXT u
9410 LET con=con+1: LET p(con)=V
AL f$
9420 PRINT BRIGHT 1;w$(1)''
9430 RETURN
```

La rutina *lee orden* consigue, indicándole la división que queremos leer, obtener todos los datos sobre la misma, es decir: número de datos, orden y nombre de la división. Para ello, el lazo de la línea 9330-9350 lee los datos de tres en tres, hasta encontrar los que buscamos. El lazo de la línea 9360-9370 lee los caracteres de la variable r$(1), que contiene la cadena *orden* y busca su final, localizando el primer blanco. El tercer lazo, líneas 9390-9400, vuelve a leer r$(1) desde el primer carácter hasta el primer blanco y, cuando encuentra una coma, asigna a la variable p( ) el valor de los caracteres numéricos contenidos en f$. De esta manera, en los sucesivos valores de p( ) estarán los valores numéricos de orden. La línea 9410 permite recuperar el último valor de orden que no lleva una coma detrás. Por último, la línea 9420 escribe el nombre de la división. Desde la línea 100 a la 999, donde termina el programa, he puesto cuatro ejemplos de cómo se podría utilizar.

```
 100 REM todos los datos
 110 REM
 120 LET dato=1: GO SUB 9300: GO
SUB 9200
 130 PAUSE 0: CLS
 140 LET dato=2: GO SUB 9300: GO
SUB 9200
 150 PAUSE 0: CLS
 160 LET dato=3: GO SUB 9300: GO
SUB 9200
 161 PAUSE 0: CLS
 162 LET dato=4: GO SUB 9300: GO
SUB 9200
 170 PAUSE 0: CLS
```

Las líneas 100 a 170 escriben los datos en la pantalla en el orden en que queramos. Así, la línea 120 asigna a la variable dato el valor 1, ejecuta las rutinas *lee orden* y *escribe* (que escribe los datos en la pantalla). La línea 130 hace una pausa y borra la pantalla para poder escribir el dato siguiente. Con un simple input esta rutina permitiría escribir en la pantalla cualquiera de los datos.

```
9200 REM escribe
9210 REM
9240 FOR s=1 TO planta
9250 FOR t=96 TO 1 STEP -1: IF a
$(p(s),t)<>" " THEN GO TO 9260
9254 NEXT t
9260 PRINT INK 2 AND s=1;a$(p(s)
,1 TO t)
9270 NEXT s
9280 RETURN
```

En la línea 220 empieza otro ejemplo de utilización del programa, *clave dicotómica,* se limpia la pantalla y se escriben en ella tres opciones. De esta manera se puede construir una clave dicotómica que va siguiendo un árbol que conduce a una determinada especie.

```
 200 REM clave dicotomica
 210 REM
 220 LET y$="": CLS : PRINT "1 "
;a$(1)'"2 ";a$(2)'"3 No sabe, no
 contesta": INPUT "¿ Cual ? (1,2
,3) :";r
 230 IF r<>1 AND r<>2 AND r<>3 T
HEN GO TO 220
 240 LET y$=("0" AND r=3)+("1" A
ND r=1)+("2" AND r=2)+y$
 250 CLS : PRINT "1 ";a$(4)'"2 "
;a$(3)'"3 No sabe, no contesta":
 INPUT "¿ Cual ? (1,2,3) :";r
 260 IF r<>1 AND r<>2 AND r<>3 T
HEN GO TO 220
 270 LET y$=("0" AND r=3)+("1" A
ND r=1)+("2" AND r=2)+y$
 998 IF y$(1)="1" AND y$(2)="1"
THEN CLS : PRINT AT 10,6;"Es una
 Angiosperma"
 999 GO TO 9999
```

Aunque en el programa sólo se señala el esqueleto de la clave de determinación, éste ofrecería otras posibilidades que no están recogidas en las claves dicotómicas normales, así permite sopesar cada respuesta e indicar al final, con respecto a la tabla introducida, la probabilidad de que la planta sea la buscada.

Para cada bifurcación hay un grupo de tres líneas del tipo de 220, 230 y 240. La 220 escribe las posibilidades de elección, la 230 evita contestaciones erróneas y la 240 genera una variable, y$, que contendrá, al final, todas las respuestas escogidas. La línea 998 es un ejemplo de cómo se puede averiguar qué respuesta es la correcta. Dado que y$ contendrá todas las respuestas, también sirve para hacer elecciones de caminos en el árbol que constituye la clave y continuar interrogando al usuario sobre una o varias de las ramas.

Por último, hemos incluido una pequeña rutina que todos los usuarios del Spectrum conocen y que sirve para poder escribir los caracteres del castellano. La rutina permite utilizar los caracteres gráficos para las letras acentuadas, la ñ y Ñ y la apertura de los signos de interrogación y admiración. Los caracteres del Spectrum están formados por una matriz de 8 × 8 «pixels».

```
9900 REM caracteres
9910 REM [redacted]
9920 DATA 4,8,56,4,60,68,60,0
9930 DATA 4,8,56,68,120,64,62,0
9940 DATA 4,8,0,24,8,8,28,0
9950 DATA 4,8,56,68,68,68,56,0
9960 DATA 4,8,68,68,68,68,56,0
9970 DATA 124,0,120,68,68,68,68,
0
9990 DATA 126,0,98,98,82,74,70,0
9991 DATA 0,16,0,16,16,16,16,0
9992 DATA 0,16,0,16,32,66,60,0
9995 RESTORE 9900: FOR s=USR "a"
 TO USR "i"+7: READ c: POKE s,c:
 NEXT s: RETURN
9999 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175:
 PRINT AT 11,6;"[redacted]"
```

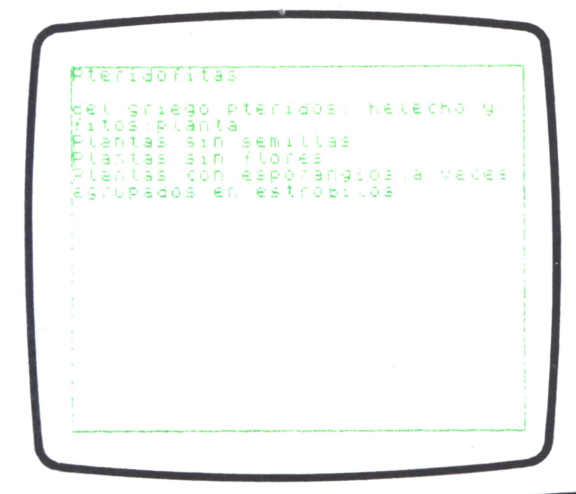
Por otro lado, sería muy fácil introducir en el programa dibujos o esquemas de partes de las plantas, lo que haría la identificación más sencilla. Para ello, al explicar el programa de dibujo de órganos hemos incluido una rutina que sería útil ahora.

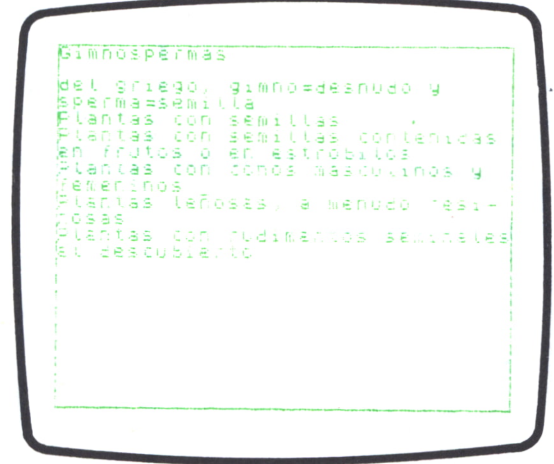
MARGA.—Bien, sólo quiero decir que en el programa, se recogen algunos nombres que son con los que normalmente se conocen a las divisiones en él incluidas; así, las Angiospermas son, en realidad, denominadas como división *Magnoliophyta,* en relación al género *Magnolia,* las Gimnos-

permas se llaman división *Pinophyta,* por el género *Pinus,* y la división *Pterydophyta,* hace referencia al género *Pteridium.*

FEDE.—Ya veo que os interesa el tema.

MARGA.—Sí, he descubierto que la botánica es un mundo apasionante...





```
Angiospermas

del griego; angios=vaso y
sperma=semilla
Plantas con semillas
Plantas con flores
Plantas con semillas contenidas
en frutos o en estrobilos
Plantas con estambres y/o pisti-
lo
Plantas herbáceas o leñosas, ge-
neralmente no resinosas
```

```
1 Plantas con semillas

2 Plantas sin semillas

3 No sabe, no contesta



 ¿ Cual ? (1,2,3...n) :1
```

```
1 Plantas con semillas

2 Plantas sin semillas

3 No sabe, no contesta



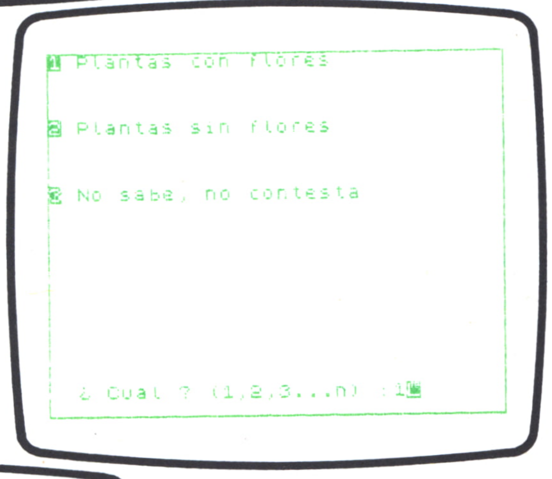
 ¿ Cual ? (1,2,3...n) :
```

1 Plantas con flores

2 Plantas sin flores

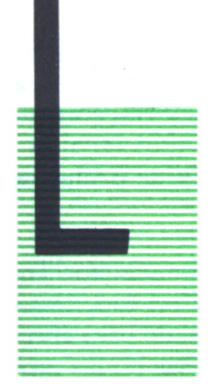
3 No sabe, no contesta

¿ Cual ? (1,2,3...n) :



Es una Angiosperma

ESTUDIO DE SECUENCIAS DE PROTEINAS 4



Jueves, 23 de abril de 1986

OS seminarios siguen su curso. Los chicos están muy animados y empiezan a atreverse con problemas más profundos.

Aunque el trabajo les ocupa la mayor parte del día, siempre encuentran tiempo para sentarse al ordenador e imaginar programas que nunca saben cómo acabar. ¿Qué les depara hoy la informática? Han decidido pasar a la acción e imbuirse en la galaxia de la bioquímica. Las proteínas, su secuenciación y la comparación de las mismas es el tema elegido. ¡Suerte!

CARLOS.—Hoy tendría que irme pronto jefe, ¿te importaría que empezásemos la reunión?

FEDE.—Si me das diez minutos para terminar esto, por mí no hay inconveniente.

CARLOS.—¿Tú puedes, Marga?

MARGA.—No he terminado este gel, pero si quieres, como no tengo prisa, lo dejo para luego.

Diez minutos después Marga y Fede están sentados frente a la pequeña pizarra de la puerta y Carlos, con el rotulador en la mano, se preparaba para comenzar su disertación.

CARLOS.—Bueno, como habíamos quedado, hoy vamos a hablar de las aplicaciones que tienen los ordenadores a la bioquímica de proteínas. En primer lugar, y como breve introducción al tema, ya que si no la hago el jefe me la va a pedir, diré que las proteínas son cadenas más o menos largas de aminoácidos, sintetizadas por los ribosomas, que son capaces de leer el ácido ribonucleico mensajero, ARNm. Este contiene la información procedente del ácido desoxirribonucleico, ADN, que se encuentra en el núcleo. Cada aminoácido es codificado a nivel del ADN por tres nucleótidos, llamados triplete. Esta información es transmitida al ARNm y leída en el citoplasma por los ribosomas que construyen las proteínas.

Por lo que he podido leer desde el jueves pasado, la secuenciación de proteínas es el campo donde antes se empezaron a utilizar los ordenadores en esta rama de la bioquímica.

MARGA.—Marchando una de historia...

CARLOS.—Vamos allá:

La historia parece que empezó con Margaret Dayhoff, que introdujo en el ordenador los datos de las escasas proteínas que hasta ese momento habían sido secuenciadas. Al mismo tiempo, y como estaba de moda en aquellos momentos, Dayhoff y sus colaboradores comenzaron a elaborar programas capaces de comparar dichas secuencias. Para estas comparaciones siguieron una técnica que no parece la mejor, pero que ha sido y sigue siendo utilizada.

FEDE.—Un momento, Carlos, me parece que vas muy de prisa, por favor, antes de comparar secuencias hay otras cosas muy importantes en la utilización de ordenadores en bioquímica de proteínas que podrías comentar.

CARLOS.—Sí, jefe, tienes razón, pero a mí la comparación de proteínas me parece apasionante.

FEDE.—Pero hay que empezar la casa por los cimientos.

CARLOS.—Daré marcha atrás. Hace años, cuando la secuenciación comenzó a ser una cosa habitual, la *IUPAC-IUB* (Commission on Biochemical Nomenclature) pensó que el código de tres letras, empleado para abreviar los aminoácidos, era muy largo al escribir las secuencias de las proteínas en los libros y revistas, y creó el código de una letra.

Así:

Alanina, que se abrevia como	Ala,	pasa a ser A,
Asparragina	Asx	B,
Cisteína	Cys	C.

De esta manera, y dado que para los bioquímicos era más fácil el código de tres letras, por ser casi inmediato, persistieron en la literatura científica ambos códigos, a pesar de las protestas de algunos, aunque la *IUPAC-IUB* recomendó el código de una letra solamente para comparar secuencias.

MARGA.—Pero el de tres letras tiene la ventaja de ser muy parecido al código genético.

Está claro que los científicos nunca se ponen de acuerdo y las sociedades parecen incapaces de escuchar a sus miembros, sobre todo cuando se trata de cosas que plantean diversidad de opiniones.

CARLOS.—Es cierto, sin embargo, para el ordenador el código de una letra sigue siendo útil, ya que, en una cadena de caracteres, cada carácter es un aminoácido y esto simplifica los programas.

En definitiva, una de las primeras cosas que conviene hacer es cons-

truir un programita capaz de transformar un código en el otro.

MARGA.—Como siempre, al final, lo que hay que hacer es otro programita. No hay manera de hacer las cosas de forma simple.

CARLOS.—Por favor, Marga, interrumpe para decir algo interesante, eres un tanto repipi.

MARGA.—Perdona, Carlos, pero me saca de quicio el trabajo inútil, y con un solo código de tres o una letra nos arreglaríamos todos mejor.

FEDE.—Pero, Marga, si tuvieses que escribir una secuencia de 1000 aminoácidos en código de tres letras opinarías de otra manera.

CARLOS.—Como resulta que la traducción de un código en otro se realiza fundamentalmente en la entrada de datos al ordenador, es importante tener en cuenta los posibles errores que el usuario del programa puede cometer y darle la posibilidad de corregirlos. Para mí, la corrección de errores en la entrada de datos, es siempre un problema.

Me gustaría que el jefe nos diese su opinión; yo he solucionado los problemas a mi manera, pero dudo que sea la mejor.

FEDE.—En realidad, cuando elaboramos un programa, siempre hacemos el intento de evitar que la persona que lo utilice pueda cometer errores y nunca se consigue. Lo mejor suele ser la experiencia del uso del mismo por gente sin idea de programación, así nos damos cuenta de lo difícil de prever qué es el ser humano y de que es capaz de cometer los errores más insospechados.

En primer lugar, el error más frecuente de un usuario que introduce secuencias de proteínas en un ordenador debe de ser el típico error de tecleo; por tanto, es importante que pueda corregir un carácter equivocado sobre la marcha.

Aquí comienzan los problemas, ya que si el error es de detección inmediata, es decir, si el usuario se da cuenta antes de introducir el aminoácido siguiente, es fácilmente subsanable, pero si se da cuenta después, ya no se sabe si es más económico tener en cuenta la posibilidad de corrección o decirle que no corrija errores de este tipo antes de terminar de introducir la secuencia, que, de todas maneras, tiene que ser cuidadosamente controlada.

Otra posibilidad frecuente es utilizar mayúsculas en vez de minúsculas, o viceversa. Esto es difícil de detectar y lo mejor, desde mi punto de vista, es que antes de que el aparato nos permita introducir el aminoácido siguiente lo busque en una tabla para ver si existe. De esta manera evitamos estos errores, pero la entrada de datos no suele ser muy rápida, porque el ordenador, cada vez que pulsas una tecla en código de una letra o tres en el de tres letras, tiene que ejecutar la rutina que busca el dato en la tabla de aminoácidos.

Otro error muy frecuente al introducir los datos es el de bailar dos aminoácidos, es decir, cambiarlos de orden, lo que supone un grave problema en la secuencia, ya que se mantiene el peso molecular de la proteína.

CARLOS.—En definitiva, como resumen, según tú, los errores cometidos pueden ser los siguientes:

1. Error de tecleo al introducir los datos.
2. Cambiar mayúsculas por minúsculas.
3. Bailar los datos.

MARGA.—Yo creo, y no es por ganas de incordiar, que se os han olvidado varias cosas que pueden suponer serios problemas. Por ejemplo, introducir dos veces el mismo aminoácido, comérselo, o cambiar uno por otro que no está en la proteína.

FEDE.—Carlos, esta niña siempre nos tiene que corregir, pero hay que reconocer que tiene toda la razón. Por ello, y ya que es más positiva que yo a la hora de elegir entre lo posible y lo práctico, dejemos que escoja la posibilidad más económica.

MARGA.—Eres muy amable; como puedes ver, me dejas lo más sencillito. Bueno, desde mi punto de vista antes de pensar en organizar una estrategia de corrección de errores te vuelvo a pasar la pelota: ¿qué tipo de gente y durante cuánto tiempo van a estar introduciendo datos?

FEDE.—Supongamos que, como no tenemos muchos duros, algo habitual, y no podemos disponer de un informático, empezamos por utilizar gente que no haya tenido experiencia con ordenadores, o alguien no relacionado con el mundo de las proteínas; una secretaria, un licenciado en letras u otra persona que no conozca el nombre de los aminoácidos y mucho menos su abreviatura.

MARGA.—¿Qué gracioso el jefito?, y por qué no piensas que estamos en USA, somos ricos y tenemos personal experto en las máquinas que continuará trabajando hasta que termine de introducir datos.

FEDE.—Sencillo, Marga, porque así es más difícil y quiero que te estrujes el cerebro un poquito, que no duele.

MARGA.—Vale, pero preferiría estrujar el tuyo. En fin, siendo como el jefe dice, lo primero que pensaría es en hacer un programa de introducción de datos lo más sencillo posible y a continuación otro para corregirlos.

El programa de introducción de datos sólo tendría la posibilidad de corregir errores de teclado y creo que haría que el ordenador comenzase por preguntar el nombre de la proteína. Sobre esto habría que hablar largo y tendido, ya que sería muy importante homogeneizar los datos introducidos.

CARLOS.—¿Qué quieres decir con homogeneizar los datos?

MARGA.—Muchas cosas; en primer lugar, habría que especificar bien qué tipo de datos son importantes en el nombre de la proteína y después crear una estructura bastante rígida para entrarlos en el ordenador. Por ejemplo, para definir la ceruloplasmina de suero de conejo sería necesario, desde mi punto de vista, decir:

animal
mamífero
lagomorfo
conejo
sangre
suero proteína ceruloplasmina

Ya que de otra manera si queremos buscar las proteínas de lagomorfos (conejos y liebres), es posible que no podamos por estar equívocamente denominadas.

CARLOS.—El orden que tú estableces no me parece el más correcto, pero entiendo lo que quieres decir.

MARGA.—Para introducir los datos de la secuencia primero habría que preguntarle al usuario si están en código de una letra o en el de tres. Creo que un ejemplo sería la mejor solución, algo como esto:

Val	His	Ala	Leu	....	(1), o así;
V	H	A	L	....	(2)

Responder 1 ó 2.

Aunque los datos pueden aparecer en los libros y las revistas de otras formas, como:

VAL	HIS	ALA	LEU	...;
Val-	His-	Ala-	Leu-	...;
val	his	ala	leu	...;
v	h	a	l	...;

Creo que estas posibilidades, como el hecho de utilizar un carácter en blanco como separador, es mejor explicárselo al que introduce los datos.

CARLOS.—De todas formas, no va a saber qué es lo que está haciendo...

MARGA.—En cuanto a la posibilidad de revisar una tabla cada vez que se introduce un aminoácido, plantea el problema de que la proteína introducida lleve en su secuencia un aminoácido raro. Esto sería un desastre, porque el ordenador se negaría a admitirlo.

Todos sabemos que existen proteínas que contienen en sus secuencias aminoácidos no comunes, como: la 4-hidroxiprolina, que se encuentra con cierta abundancia en la proteína fibrosa colágeno y en algunas proteínas de las plantas; la hidroxilina, que es un derivado de la lisina y también aparece en el colágeno, o la desmosina y la isodesmosina, que se encuentran en la elastina.

FEDE.—Bravo Marga, ves, en esa posibilidad yo no había pensado y es muy real. También es importante considerar las formas acetiladas de los aminoácidos que aparecen bastante frecuentemente como primer residuo.

MARGA.—Gracias, jefe, una que es inteligente. El final del primer programa sería sencillo. Por ejemplo:

Para terminar la introducción de una secuencia pulse espacio y guión.

Desde mi punto de vista, una sucesión rara de caracteres siempre es más útil que el simple RETURN, ya que tendemos a pulsarlo muchas veces por equivocación. Sin embargo, si no se recuerda esta sucesión de caracteres se puede recurrir a la pantalla de ayuda, que debería de existir siempre, y ser corregida y ampliada por el programador a medida que distintos usuarios utilizan el programa.

CARLOS.—Entonces la pantalla será infinita, ocupará toda la memoria del ordenador y nunca más podrá ser utilizado en pos de la consideración humana.

El programa que he hecho es el siguiente:

```
   1 REM rutina principal
   2 REM ================
  10 LET s$="": LET t$="": LET p
m=0: LET naa=0: LET num=0: LET b
$="
    ": GO SUB 8000
  20 INPUT "Los datos estan en c
odigo de     tres o de una letra
(3,1): ";a
  30 IF a=3 THEN PRINT AT 0,0;b$
: GO SUB 1000
  40 IF a=1 THEN PRINT AT 0,0;b$
: GO SUB 3000
  50 IF a<>3 AND a<>1 THEN PRINT
 #0,AT 0,0;b$;AT 0,0;"Error, ";a
;" no es ni 1 ni 3 ": PAUSE 0: G
O TO 20
  60 GO TO 9000
1000 REM entrada de datos en
          codigo de tres letras
1010 REM =====================
1020 INPUT LINE a$: LET num=num+
1: GO SUB 2000
1030 IF a$="Fin" THEN LET num=nu
m-1: GO TO 1110
1040 IF LEN a$<>3 THEN PRINT #0;
"Error no tiene los 3 caracteres
": LET num=num-1: BEEP 0.05,25:
PAUSE 0: GO TO 1020
1050 FOR s=1 TO 26
1060 IF k$(s)=a$ THEN LET s$=s$+
("-" AND LEN s$>0)+a$: LET pm=pm
+l(s)-18.016: LET m(s)=m(s)+1: L
ET t$=t$+l$(s): GO TO 1090
1070 NEXT s
1080 PRINT #0;"Error, no conozco
 ese aminoacido": BEEP 0.05,20:
PAUSE 0: LET num=num-1*(num>0):
GO TO 1020
1090 CLS : PRINT s$;"-";
1100 GO TO 1020
1110 RETURN
2000 REM mayusculas a minusculas
2010 REM ======================
2020 IF CODE a$(1)>90 THEN LET a
$(1)=CHR$ (CODE (a$(1))-32)
2030 IF LEN a$=1 THEN RETURN
2040 FOR t=2 TO 3
2050 IF CODE a$(t)<CODE "Z" THEN
 LET a$(t)=CHR$ (CODE a$(t)+32)
2060 NEXT t
2070 RETURN
```

```
3000 REM entrada de datos en
        codigo de una letra
3010 REM ===================
3020 INPUT LINE a$: LET num=num+
1
3030 IF a$="Fin" THEN RETURN
3040 IF LEN a$<>1 THEN PRINT #0;
"Error, solo quiero 1 caracter":
 PAUSE 0: GO TO 3020
3050 GO SUB 2000
3060 LET aa=((CODE a$)-64): IF l
$(aa)=a$ THEN LET s$=s$+("-" AND
 LEN s$>2)+k$(aa): LET pm=pm+l(a
a)-18.016: LET m(aa)=m(aa)+1: LE
T t$=t$+("-" AND LEN t$<>0)+l$(a
a)
3070 PRINT AT 0,0;s$
3080 GO TO 3020
8500 REM Datos de aminoacidos
8510 REM ====================
8520 DATA "Ala","A",89.09: REM
 Alanina
8530 DATA "Asx","B",132.61: REM
Ac. Aspartico o Asparragina
8540 DATA "Cys","C",121.15: REM
Cisteina
8550 DATA "Asp","D",133.10: REM
Acido Aspartico
8560 DATA "Glu","E",147.13: REM
Acido Glutamico
8570 DATA "Phe","F",165.19: REM
Fenilalanina
8580 DATA "Gly","G",75.07: REM
Glicocola
8590 DATA "His","H",155.16: REM
histidina
8595 DATA "Ile","I",131.17: REM
Isoleucina
8600 DATA "Noa","J",00.00: REM
        No aminoacido
8610 DATA "Lys","K",146.19: REM
Lisina
8620 DATA "Leu","L",131.17: REM
Leucina
8630 DATA "Met","M",149.21: REM
Metionina
8640 DATA "Asn","N",132.12: REM
Asparraguina
8650 DATA "Noa","O",00.00: REM
        No aminoacido
8660 DATA "Pro","P",115.13: REM
Prolina
8670 DATA "Gln","Q",146.15: REM
Glutamina
8680 DATA "Arg","R",174.20: REM
Arginina
8690 DATA "Ser","S",105.09: REM
Serina
8700 DATA "Thr","T",119.12: REM
Treonina
8710 DATA "Noa","U",00.00: REM
        No aminoacido
8720 DATA "Val","V",117.15: REM
Valina
8730 DATA "Trp","W",204.23: REM
Triptofano
8740 DATA "Aar","X",111.00: REM
        Aminoacido raro
8750 DATA "Tyr","Y",181.19: REM
Tirosina
8760 DATA "Glx","Z",146.64: REM
Ac. Glutamico o Glutamina
8770 DIM k$(26,3): DIM l$(26)
8780 DIM l(26): DIM m(26): RESTO
RE 8520
```

```
8790 FOR s=1 TO 26
8800 READ k$(s),l$(s),l(s)
8810 NEXT s
8820 RETURN
9000 REM escribe resultados
9010 REM ==================
9012 LET i=2: INPUT "Impresora o
 pantalla (i,p): ";j$: IF j$="i"
 THEN LET i=3
9020 LET pm=pm+18.016: CLS : PRI
NT #i;"Peso Molecular= ";pm
9030 PRINT #i;''"Composicion"''
: FOR s=1 TO 26: PRINT #i;k$(s);
" ";m(s);" ";
9040 IF s/5=INT (s/5) THEN PRINT
 #i
9050 NEXT s
9060 PRINT AT 21,0;"pulse una te
cla": PAUSE 0
9070 CLS : PRINT #i;''"SECUENCIA
"''t$''s$
9100 REM guarda secuencia
9110 REM ================
9120 INPUT "La quiere guardar (s
,n): ";g$
9130 IF g$<>"s" THEN GO TO 9999
9140 INPUT "Nombre de la protein
a: ";m$: LET n$=m$(1 TO 8)+".S"
9150 LET o$="": FOR s=1 TO 26: L
ET o$=o$+STR$ (m(s))+"": NEXT s
9160 LET s$=n$+"*"+STR$ pm+"|"+o
$+"\"+s$+"/"+t$
9170 SAVE *"m";1;n$: VERIFY *"m"
;1;n$
```

El programita, como veis, está hecho en el Spectrum y he comprobado que es bastante fiable.

```
   1 REM rutina principal
   2 REM ================
  10 LET s$="": LET t$="": LET p
m=0: LET naa=0: LET num=0: LET b
$="
  ": GO SUB 8000
  20 INPUT "Los datos estan en c
odigo de    tres o de una letra
(3,1): ";a
  30 IF a=3 THEN PRINT AT 0,0;b$
: GO SUB 1000
  40 IF a=1 THEN PRINT AT 0,0;b$
: GO SUB 3000
  50 IF a<>3 AND a<>1 THEN PRINT
 #0,AT 0,0;b$;AT 0,0;"Error, ";a
;" no es ni 1 ni 3 ": PAUSE 0: G
O TO 20
  60 GO TO 9000
```

La línea 10 genera una serie de variables que van a ser utilizadas en el programa y que necesitan un valor inicial.

La línea 20 permite definir si los datos están en código de una o de tres letras, y a continuación las líneas 30 y 40 pasan la ejecución a la rutina correspondiente para entrar cada tipo de datos.

Si el programa llega a la línea 50, será debido a que la respuesta dada en la 20 no es ni 1 ni 3, en cuyo caso el error se corrige volviendo a la línea 20, permitiéndote empezar de nuevo.

FEDE.—Y poner más cuidado la próxima vez.

```
1000 REM entrada de datos en
         codigo de tres letras
1010 REM ====================
1020 INPUT LINE a$: LET num=num+
1: GO SUB 2000
1030 IF a$="Fin" THEN LET num=nu
m-1: GO TO 1110
1040 IF LEN a$<>3 THEN PRINT #0;
"Error no tiene los 3 caracteres
": LET num=num-1: BEEP 0.05,25:
PAUSE 0: GO TO 1020
1050 FOR s=1 TO 26
1060 IF k$(s)=a$ THEN LET s$=s$+
("-" AND LEN s$>0)+a$: LET pm=pm
+l(s)-18.016: LET m(s)=m(s)+1: L
ET t$=t$+l$(s): GO TO 1090
1070 NEXT s
1080 PRINT #0;"Error, no conozco
 ese aminoacido": BEEP 0.05,20:
PAUSE 0: LET num=num-1*(num>0):
GO TO 1020
1090 CLS : PRINT s$;"-";
1100 GO TO 1020
1110 RETURN
```

CARLOS.—La rutina *entrada de datos en código de tres letras* empieza en la línea 1000 y termina en la línea 1110. Sirve para introducir este tipo de datos y traducirlos a código de una letra. Las variables s$ y t$, a las que se da un valor inicial en la línea 10, servirán para contener la secuencia completa introducida, en códigos de tres y una letra, respectivamente.

La rutina comienza en la línea 1020 pidiendo un dato, poniendo en marcha el contador de aminoácidos introducidos, número, y ejecutando la rutina *minúsculas a mayúsculas.* Este fragmento de programa permite, entrando en la línea 2020, transformar los dos últimos caracteres en minúsculas controlando, a su vez, que el primero sea mayúscula para evitar los errores comunes de las distintas formas de escribir los aminoácidos en código de tres letras: Ala, ALA, ala, por alanina.

La línea 1030 comprueba que el usuario no ha escrito Fin y, por tanto, no quiere terminar de introducir datos.

MARGA.—Y cómo se escribe, FIN, fin o Fin.

CARLOS.—Graciosilla, je, je..., en este caso admite las tres.

La línea 1040 comprueba que se han escrito tres caracteres y si no es así indica que se ha cometido un error, hace sonar un pitido y descontando un aminoácido al contador, num, vuelve 1020.

La 1050 es el comienzo de un lazo que revisa la tabla de aminoácidos introducida en k$ (línea 10, GOSUB 8770) y permite localizar el que hemos escrito en a$. Para ello, la línea 1060 compara a$ con k$ y, si en algún momento son iguales, coloca este aminoácido en dos cadenas de caracteres que contendrán las secuencias en códigos de una o tres letras, respectivamente, s$ y t$.

Al mismo tiempo, este lazo va acumulando en pm el peso molecular de la proteína, al ir sumando el peso molecular del residuo en cuestión que está en l() y restándole una molécula de agua a cada nuevo aminoácido incluido. La línea 9020 suma la molécula de agua de más que al finalizar la secuencia resta este lazo.

FEDE.—Correcto, para calcular el peso molecular de la proteína se suman los de sus aminoácidos y se restan tantas moléculas de agua como de aminoácidos tenga menos una, ya que se pierde una molécula de agua por enlace peptídico formado.

CARLOS.—También se va construyendo en m() una tabla que contendrá el número de veces que cada aminoácido aparece en la cadena. Esta última tabla, que ya Dayhoff incluye en sus publicaciones, es muy útil para comprobar que la secuencia se ha escrito correctamente. En la cadena s$, en código de tres letras, el computador pone un guión de separación entre los aminoácidos excepto en el primer aminoácido, evitando así que la secuencia comience por un guión.

Si se ejecuta la línea 1080 quiere decir que el lazo anterior terminó sin que apareciese el residuo que buscábamos. El ordenador emite un sonido, BEEP, y vuelve a preguntar.

La línea 1090 muestra en la pantalla la secuencia que hemos escrito y un guión para separarla. La siguiente, 1100, vuelve a la 1020 para dejarnos escribir el siguiente aminoácido. La línea 1110 termina la rutina con RETURN.

MARGA.—¡Ah, es verdad!, y así se evita el bailar los aminoácidos inadvertidamente al verlos en la pantalla.

```
3000 REM entrada de datos en
          codigo de una letra
3010 REM ==================
3020 INPUT LINE a$: LET num=num+
1
3030 IF a$="Fin" THEN RETURN
3040 IF LEN a$<>1 THEN PRINT #0;
"Error, solo quiero 1 caracter":
 PAUSE 0: GO TO 3020
3050 GO SUB 2000
3060 LET aa=((CODE a$)-64): IF l
$(aa)=a$ THEN LET s$=s$+("-" AND
 LEN s$>2)+k$(aa): LET pm=pm+l(a
a)-18.016: LET m(aa)=m(aa)+1: LE
T t$=t$+("-" AND LEN t$<>0)+l$(a
a)
3070 PRINT AT 0,0;s$
3080 GO TO 3020
```

```
8500  REM Datos de aminoàcidos
8510  REM ===================
8520  DATA  "Ala",  "A",   89.09: REM Alanina
8530  DATA  "Asx",  "B",  132.61: REM Acido aspàrtico o Asparragina
8540  DATA  "Cys",  "C",  121.15: REM Cisteina
8550  DATA  "Asp",  "D",  133.10: REM Acido aspàrtico
8560  DATA  "Glu",  "E",  147.13: REM Acido glutàmico
8570  DATA  "Phe",  "F",  165.19: REM Fenilalanina
8580  DATA  "Gly",  "G",   75.07: REM Glicocola
8590  DATA  "His",  "H",  155.16: REM Histidina
8595  DATA  "Ile",  "I",  131.17: REM Isoleucina
8600  DATA  "Noa",  "J",   00.00: REM No aminoàcido
8610  DATA  "Lys",  "K",  146.19: REM Lisina
8620  DATA  "Leu",  "L",  131.17: REM Leucina
8630  DATA  "Met",  "M",  149.21: REM Metionina
8640  DATA  "Asn",  "N",  132.12: REM Asparragina
8650  DATA  "Noa",  "O",   00.00: REM No aminoàcido
8660  DATA  "Pro",  "P",  115.13: REM Prolina
8670  DATA  "Gln",  "Q",  146.15: REM Acido Glutàmico
8680  DATA  "Arg",  "R",  174.20: REM Arginina
8690  DATA  "Ser",  "S",  105.09: REM Serina
8700  DATA  "Thr",  "T",  119.12: REM Treonina
8710  DATA  "Noa",  "U",  000.00: REM No aminoàcido
8720  DATA  "Val",  "V",  117.15: REM Valina
8730  DATA  "Trp",  "W",  204.23: REM Asparragina
8740  DATA  "Aar",  "X",  111.00: REM Aminoàcido raro
8750  DATA  "Tyr",  "Y",  181.19: REM Tirosina
8760  DATA  "Glx",  "Z",  146.64: REM Acido Glutàmico o Glutamina
8770  DIM k$(26,3): DIM l$(26)
8780  DIM l(26): DIM m(26): RESTORE 8520
8790  FOR s = 1 TO 26
8800  READ k$(s), l$(s), l(s)
8810  NEXT s
8820  RETURN
```

CARLOS.—La rutina de *entrada de datos en código de una letra,* que empieza en 3000, permite, en 3020, la entrada de un aminoácido y suma 1 al contador num. La línea 3040 comprueba que solamente se haya escrito un carácter. En caso contrario, el programa vuelve a 3020 preguntando otro carácter.

La línea 3030 permite terminar la entrada de datos escribiendo Fin, y esta vez sólo Fin es válido.

MARGA.—¿Todos los datos que entramos pasan por la rutina de transformación de mayúsculas a minúsculas?

CARLOS.—¡Claro, mujer!

```
2000 REM mayusculas a minusculas
2010 REM ====================
2020 IF CODE a$(1)>90 THEN LET a
$(1)=CHR$ (CODE (a$(1))-32)
2030 IF LEN a$=1 THEN RETURN
2040 FOR t=2 TO 3
2050 IF CODE a$(t)<CODE "Z" THEN
 LET a$(t)=CHR$ (CODE a$(t)+32)
2060 NEXT t
```

En la línea 2000 comienza la rutina *mayúsculas a minúsculas,* que se encarga de traducir los caracteres al formato de la tabla de aminoácidos. Para ello, la línea 2020 comprueba si el primer carácter escrito es mayor que el carácter *ASCII* 123, que es la z minúscula. Si lo es, resta 32 a este valor, con la esperanza de que el resultado sea la mayúscula correspondiente. Es interesante hacer notar que el código *ASCII* (American Standard Code for Interchange Information) está escrito de manera que esto se cumpla (f = 102 y F = 70).

```
8500  REM Datos de aminoàcidos
8510  REM ====================
8520  DATA  "Ala", "A",  89.09: REM Alanina
8530  DATA  "Asx", "B", 132.61: REM Acido aspàrtico o Asparragina
8540  DATA  "Cys", "C", 121.15: REM Cisteina
8550  DATA  "Asp", "D", 133.10: REM Acido aspàrtico
8560  DATA  "Glu", "E", 147.13: REM Acido glutàmico
8570  DATA  "Phe", "F", 165.19: REM Fenilalanina
8580  DATA  "Gly", "G",  75.07: REM Glicocola
8590  DATA  "His", "H", 155.16: REM Histidina
8595  DATA  "Ile", "I", 131.17: REM Isoleucina
8600  DATA  "Noa". "J",  00.00: REM No aminoàcido
8610  DATA  "Lys", "K", 146.19: REM Lisina
8620  DATA  "Leu", "L", 131.17: REM Leucina
8630  DATA  "Met", "M", 149.21: REM Metionina
8640  DATA  "Asn", "N", 132.12: REM Asparragina
8650  DATA  "Noa", "O",  00.00: REM No aminoàcido
8660  DATA  "Pro", "P", 115.13: REM Prolina
8670  DATA  "Gln", "Q", 146.15: REM Acido Glutàmico
8680  DATA  "Arg", "R", 174.20: REM Arginina
8690  DATA  "Ser", "S", 105.09: REM Serina
8700  DATA  "Thr", "T", 119.12: REM Treonina
8710  DATA  "Noa", "U", 000.00: REM No aminoàcido
8720  DATA  "Val", "V", 117.15: REM Valina
8730  DATA  "Trp", "W". 204.23: REM Asparragina
8740  DATA  "Aar", "X", 111.00: REM Aminoàcido raro
8750  DATA  "Tyr", "Y", 181.19: REM Tirosina
8760  DATA  "Glx", "Z", 146.64: REM Acido Glutàmico o Glutamina
8770  DIM k$(26,3): DIM l$(26)
8780  DIM l(26): DIM m(26): RESTORE 8520
8790  FOR s = 1 TO 26
8800  READ k$(s), l$(s), l(s)
8810  NEXT s
8820  RETURN
```

Por último, la rutina que comienza en la línea 8500, *datos de aminoácidos,* contiene los datos de los 20 aminoácidos fundamentales. Casi todas las proteínas conocidas hasta ahora contienen únicamente estos 20 aminoácidos, pero algunas presentan aminoácidos modificados ligeramente u otros raros. Cada aminoácido está representado en la tabla en código de una letra, de tres, y por su peso molecular.

Como en código de una letra casi todas representan un aminoácido, los he puesto por orden alfabético. La letra que no tiene aminoácido asig-

nado aparece Noa, por NO Aminoácido, menos en la X que he dejado para cualquiera que no esté en la lista, asignándole el peso molecular de 111, por consejo del jefe.

FEDE.—Tengo que puntualizar que aunque la media del peso molecular de los 20 aa es de aproximadamente 140, el número 111 se acerca más al valor obtenido dividiendo el peso molecular de varias proteínas por su número de aminoácidos, indicando que son más frecuentes los aminoácidos de bajo peso molecular. Este número ha sido obtenido utilizando proteínas solubles, pero no tiene por qué ser válido para todas, y de hecho Dayhoff utiliza algo así como 128.

CARLOS.—Como siempre, el jefe no puntualizó tanto cuando le pregunté, pero, en fin, tengo que reconocer que tiene razón y ahora pienso que, quizá, es mejor en este caso utilizar la media.

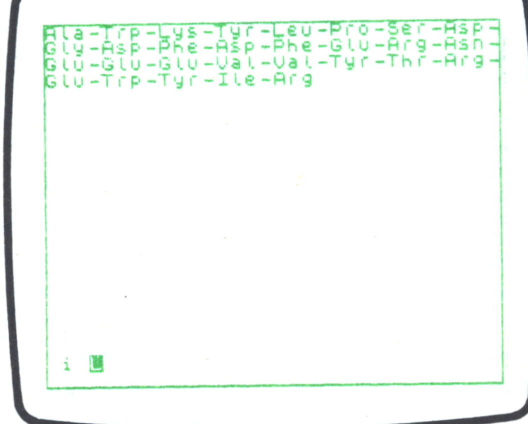
Bueno, sigo, las líneas 8770 a 8820 permiten cargar los datos en las variables; en k$() estarán los códigos de tres letras, en l$() los de una, en l() los pesos moleculares y en m() una tabla que contendrá el número de veces que aparece cada aa en la secuencia. El **RESTORE** de la línea 8780 permite que siempre empiece a leer por el data de la línea 8520 y de esta manera el read de la lína 8800 lea primero la Ala, A, 89,10 y los escriba en k$(1), l$(1) y l(1), y así sucesivamente.

```
9000 REM escribe resultados
9010 REM ==================
9012 LET i=2: INPUT "Impresora o
 pantalla (i,p): ";j$: IF j$="i"
 THEN LET i=3
9020 LET pm=pm+18.016: CLS : PRI
NT #i;"Peso Molecular= ";pm
9030 PRINT #i;''"Composicion"''
: FOR s=1 TO 26: PRINT #i;k$(s);
" ";m(s);" ",
9040 IF s/5=INT (s/5) THEN PRINT
 #i
9050 NEXT s
9060 PRINT AT 21,0;"pulse una te
cla": PAUSE 0
9070 CLS : PRINT #i;''"SECUENCIA
:"''t$''s$
9100 REM guarda secuencia
9110 REM =================
9120 INPUT "La quiere guardar (s
,n): ";g$
9130 IF g$<>"s" THEN GO TO 9999
9140 INPUT "Nombre de la protein
a: ";m$: LET n$=m$(1 TO 8)+".S"
9150 LET o$="": FOR s=1 TO 26: L
ET o$=o$+STR$ (m(s))+"*": NEXT s
9160 LET s$=n$+"*"+STR$ pm+"|"+o
$+"\"+s$+"/"+t$
9170 SAVE *"m";1;n$: VERIFY *"m"
;1;n$
```

El programa contiene otras dos rutinas: *escribe resultados* y *guarda secuencias*. La primera escribe los resultados del programa: peso molecular de la proteína, composición de aminoácidos y secuencia en códigos de una

Los datos estan en codigo de
tres o de una letra (3,1):

Error, 11 no es ni 1 ni 3



```
Trp-Asp-Asn-Leu-Pro-Phe-Lys-Leu-
Glu-Leu-

 Error,no conozco el aminoacido
```

```
Peso Molecular= 5728.182

Composicion=

Ala 2 Asx 0 Cys 1 Asp 3 Glu 8
Phe 6 Gly 4 His 1 Ile 1 Noa 0
Lys 4 Leu 3 Met 0 Asn 1 Noa 0
Pro 2 Gln 4 Arg 2 Ser 2 Thr 3
Noa 0 Val 1 Trp 0 Aar 0 Tyr 1
Glx 0

Secuencia=

TSEDHFQPFFNEKTFGAGEADCGLRPLFEKKQ
VQDETQKELFESYIEGR

Thr-Ser-Glu-Asp-His-Phe-Gln-Pro-
Phe-Phe-Asn-Glu-Lys-Thr-Phe-Gly-
Ala-Gly-Glu-Ala-Asp-Cys-Gly-Leu-
Arg-Pro-Leu-Phe-Glu-Lys-Lys-Gln-
Val-Gln-Asp-Glu-Thr-Gln-Lys-Glu-
Leu-Phe-Glu-Ser-Tyr-Ile-Glu-Gly-
Arg
```

y tres letras. La rutina sólo tiene una instrucción ligeramente problemática, que es cuando permite imprimir o ver los resultados en la pantalla. La rutina *guarda secuencia* construye una cadena de caracteres que contiene todos los valores que hemos obtenido con el programa. Para construir esta cadena los datos, transformados en caracteres, se suman y se separan utilizando un símbolo raro entre dato y dato para poder recuperarlos.

MARGA.—¡Uf! Creía que no ibas a terminar nunca.

FEDE.—Explicar un programa es algo complejo; quien lo construye entiende cómo funciona, pero no siempre es capaz de hacerlo entender a los demás.

CARLOS.—Continúo...

Podéis ver que mi programa utiliza alguno de los sistemas de corrección de errores que vosotros habéis citado, pero otros ni los habías olido. Para poder introducir aminoácidos raros me he inventado un nuevo aminoácido, el Aar, es decir, AminoAcido Raro, o X.

Por otra parte, y para evitar el error en las diferentes maneras de escribir los aminoácidos he optado por forzar a que la primera letra del código de tres tenga que ser mayúscula y las otras dos minúsculas, ya que ésta es la forma más corriente de escribirlos. De esta manera se suprime el problema de los formatos, ya que se pueden introducir los datos como se quiera en cuanto a mayúsculas y minúsculas. También he tratado de evitar las pulsaciones defectuosas que provocarían el que el aminoácido introducido no tenga tres caracteres o uno en cada caso.

Por último, pensado en la manera mejor de comprobar de un vistazo si la secuencia es correctamente introducida, el programa tiene una rutina que imprime al final la fórmula resumida de la proteína, como, por otra parte, hace Dayhoff, ya que una tabla que señale el número de veces que cada aminoácido aparece en la secuencia es indicio de que está correctamente introducida.

Marga, esto es un reto...

MARGA.—Acepto.

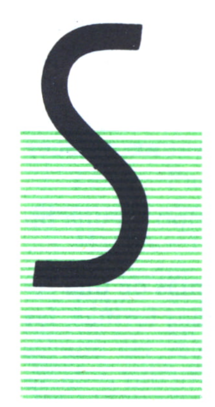
CARLOS.—Sencillo, ¿verdad? No creo que pueda haber un pograma más corto y más completo; además, las rutinas ofrecen un montón de posibilidades.

Lo que destaca es la posibilidad de utilizar una tabla con ambos códigos. Es importante para la velocidad de transformación de un código en otro el hecho de que, dado que el código de una letra es el alfabeto, se ha utilizado el valor *ASCII* de los caracteres para buscar directamente en la tabla la traducción de código de una letra a código de tres letras.

MARGA.—Eso ya lo has dicho antes...

OTRAS APLICACIONES 5

Jueves, 12 de junio de 1986



E acerca el verano, parece que ésta sea una época de balance. En el laboratorio se hace resumen de los resultados obtenidos, se ultiman experimentos y se piensa en cuáles pueden ser los siguientes. Marga ya es químico y la tesis de Carlos empieza a adquirir forma.

Ha sido un año duro para todos, pero el doctor Ruiz está contento, sus pupilos son ya capaces de afrontar los problemas cotidianos sin necesidad de andar siempre desangelados por el laboratorio. El es para ellos un compañero más experto que puede ayudarles cuando lo necesitan. También han conseguido adentrarse sin miedo en el mundo de la informática, saben que aún están muy lejos de comprenderlo todo, pero no les preocupa, han empezado con buen pie.

Hoy el seminario será algo especial, no habrá programas que discutir ni que probar en el ordenador. Quieren saber otras cosas.

FEDE.—¿Qué os parece si el seminario de hoy lo hacemos fuera del laboratorio?

CARLOS.—¡Propongo el césped! Es el lugar más idóneo.

FEDE.—Tienes razón, pero mi propuesta es otra, podemos darnos un paseo por otros centros relacionados con la biología o la medicina y ver qué utilización hacen allí de la informática.

MARGA.—¡Pues nos va a llevar mucho tiempo!, ya que hoy es raro el laboratorio o departamento que no tiene un ordenador.

FEDE.—Ya lo sé, sólo quiero que nos hagamos una idea de para qué le sirve a un científico, siempre podemos encontrar aplicaciones que no se nos hayan ocurrido.

Salen del laboratorio y se dirigen hacia el aparcamiento. Al cabo de media hora se encuentran en la calle Joaquín Costa, a la altura de la plaza de la República Argentina.

MARGA.—Fede, ¿dónde estamos?

FEDE.—Estamos en el *ICYT*, uno de los centros del *CSIC*.

CARLOS.—¡Vaya lío de siglas!

FEDE.—*CSIC* significa Centro Superior de Investigaciones Científicas, e *ICYT*, Instituto de Información y Documentación en Ciencia y Tecnología.

Entremos.

CARLOS.—¿Se puede saber qué hemos venido a hacer aquí, Fede? Nos tiene intrigados.

FEDE.—Como sabéis, en nuestra profesión hay que estar siempre al día, y para ello tienes que conocer todo lo que se hace en el resto del mundo, al menos en el tema en que trabajas. Existen bancos de datos en los que queda recogida la mayor parte de los artículos publicados en revistas científicas a lo largo de los años; si quieres hacer una revisión bibliográfica completa, puedes acceder a ellos de varias formas, lo que vamos a hacer nosotros es una de ellas.

MARGA.—Esto es mucho más cómodo que pasarse el tiempo en una biblioteca.

FEDE.—Tienes razón, en las bibliotecas se suele escapar más de una referencia que puede contener cosas muy interesantes. De todas formas, las bibliotecas son útiles en este caso, ya que permiten conocer las nuevas tendencias de la ciencia, ver cómo está estructurado un artículo o darse cuenta de la gran cantidad de revistas que existen.

CARLOS.—¿Y cómo funciona?

FEDE.—Muy sencillo, eliges el tema que te interesa, especificando palabras o frases que estén directamente relacionadas con el mismo o al contrario, qué aspectos no son de tu interés. La consulta la hacen por teléfono a través del ordenador conectado con el del banco de datos. El operador del ordenador del *ICYT* llama al banco de datos y se conecta, a través del «MODEM», con el del banco. Una vez realizada la conexión, el que maneja el ordenador del *ICYT* comienza preguntando las palabras clave que tú le has dado, a lo que el banco de datos le responde con el número de artículos que sobre este tema tiene almacenados. A continuación se hacen todas las elecciones pertinentes y se da por terminada la conexión. Al cabo de unos días recibes un listado con los artículos que el ordenador ha encontrado, indicando revista, autores, título y un breve resumen del mismo.

CARLOS.—Pero algunos temas son muy tratados y el número de artículos sobre ellos es inmenso. Imagínate que le pides todas las referencias que existen sobre proteínas.

MARGA.—Pues estaría listando toda la vida.

FEDE.—Claro, pero tú puedes restringir la búsqueda tanto como quieras, incluyendo varias palabras que la delimiten o mezclando las elecciones.

Por ejemplo:

SET	ITEMS	DESCRIPTION
1	1345	CONCANAVALINA A
2	23	FAVIN
3	202140	CELL
4	12	1*2
5	4	3*4

SET = número que indica una búsqueda.

ITEMS = número de artículos que contienen esta palabra.

DESCRIPTION = descripción de la palabra o de la búsqueda.

Sobre la concanavalina hay 1.345 artículos y 23 de la fabina, 202.140 de células y 12 que contienen fabina y concanavalina. De los 12 artículos, 4 tienen también la palabra células.

CARLOS.—¿Y si las referencias están escritas en distintos idiomas?

FEDE.—Pues has de escribir la palabra de todas las formas posibles o indicar qué idiomas son los que te interesa manejar.

MARGA.—Imagino que habrá otras formas de ponerse en contacto con el ordenador.

FEDE.—Sí que las hay; puedes hacerlo sin moverte del laboratorio, a través de un «MODEM» acoplado al teléfono. La consulta se haría de igual manera y cada día son más los laboratorios que tienen contacto directo con los ordenadores centrales de la base de datos.

CARLOS.—¿Dónde están estos ordenadores?

FEDE.—Hay varios, uno de ellos se encuentra en Italia, otro en Estados Unidos.

CARLOS.—¿En España no hay ninguno?

FEDE.—No. Bueno, ya que tenemos los impresos rellenos podemos hacer la consulta y esperar a que nos llegue el listado.

Después deciden acercarse a la Facultad de Ciencias Biológicas y pasear por sus departamentos.

FEDE.—Esto te resulta familiar, ¿verdad Carlos?

CARLOS.—Claro, cinco años no se olvidan fácilmente.

FEDE.—Pues puedes hacernos de guía, ¿no te parece?

CARLOS.—De acuerdo. Vamos a ello.

El proceso de informatización de la Universidad es lento, pero progresivo, en cualquier facultad se usan los ordenadores como elemento de trabajo y los biólogos son conscientes de su importancia. No sólo los que estudian Telecomunicaciones, Informática o Cálculo Automático los usan, sino que todos los universitarios empiezan a hacerlo.

La Universidad dispone de centros de cálculo abiertos a todos aquellos que los necesiten, así se puede manejar el ordenador central. No sólo existen estos ordenadores grandes, sino que hay muchos otros ordenadores personales en los distintos departamentos y cátedras que tienen un uso

continuado, incluso por parte de los alumnos. Eso sí, tienes que estar en los últimos cursos o haciendo una tesis, pero, bueno...

Así se utilizan en bioquímica, como ya sabemos; en genética, para estudios de poblaciones, en su rama cuantitativa o en genética evolutiva; en antropología, en estudios de grupos sanguíneos, determinación de caracteres raciales o variación de las medidas durante el crecimiento; en ecología, para distribución de especies, evolución de ecosistemas o asociación de los diferentes elementos que constituyen el medio ambiente. Podría seguir enumerando y creo que no acabaría nunca.

No sólo se utilizan con fines meramente estadísticos, sino que son útiles a otros muchos niveles, como el educativo. Por ejemplo, una universidad americana utiliza un determinado tipo de ordenador personal para los alumnos; tiene actualmente 10.000 de estos aparatos y espera tener más.

Lo bueno de este sistema es que están elaborando software muy sofisticado, y así, en su Departamento de Químicas, tienen un editor molecular que puede representar en la pantalla 100 átomos diferentes de la tabla periódica, diversas estructuras químicas básicas y grupos funcionales. El alumno puede dibujar en la pantalla moléculas diferentes y fundir parte de ellas, medir distancias interatómicas, el ángulo entre los enlaces y otros muchos datos atómicos.

En el Departamento de Historia han elaborado otro programa que no sólo permite representar los movimientos de tropas en la Primera Guerra Mundial, sino que permite colocarse en la piel de los políticos y participar en la división de fronteras que tuvo lugar al firmarse la paz.

MARGA.—Es fantástico, y lo que hace falta es que todos estos programas lleguen a las aulas aquí, que no sólo estén en los departamentos o en los centros de cálculo, sino que todos los días se pueda trabajar con ellos, no de forma ocasional y tras esperar varias horas ante un terminal o en la cola de una perforadora de tarjetas.

FEDE.—Es cierto, pero poco a poco ocurrirá lo que tú dices. Empieza ya a haber universidades que disponen de un ordenador para cada alumno con programas como los que ha comentado Carlos y que tienen gabinetes de especialistas en informática para solucionar cualquier duda que pueda surgir durante el estudio.

MARGA.—Una especie de «hot line»...

FEDE.—Algo parecido, así, si surge algún problema, siempre sabes a quién recurrir y no tienes por qué esperar hasta la próxima clase.

CARLOS.—Me gustaría saber cuándo se empezaron a utilizar los ordenadores en el mundo de la ciencia, ya sé que en 1975 Dayhoff los utilizaba en bioquímica, pero me imagino que antes habría programas destinados al mundo de la biología o de las ciencias naturales en general.

FEDE.—No puedo precisar mucho sobre este tema, pero, por ejemplo, ya en 1966 existían programas de simulación dentro de la paleontología;

éstos reproducen el proceso de torsión que ha sufrido la concha de los gasterópodos a lo largo de la evolución y también son de esa década programas de digitalización de cromosomas, dentro de la genética, lo que permite comparar unos cariotipos con otros y detectar cualquier tipo de mutación.

MARGA.—Ya veo que desde hace mucho tiempo se confía en los ordenadores, y que se les sabe sacar partido para cualquier tipo de aplicación. Creo que llevamos especificadas un buen número de ellas.

FEDE.—Pero, como dicen en los circos, aún hay más.

Vuelven al laboratorio. En los pasillos de su departamento aparecen alineados gran cantidad de aparatos, cada uno con una función diferente. Marga y Carlos observan que la mayoría de ellos tienen una pantalla al lado y un teclado, en todos hay también una pequeña impresora que lista páginas y páginas sin descanso.

CARLOS.—Todo esto son ordenadores, ¿no?

MARGA.—¿Cómo?, no es posible.

FEDE.—Sí, todos los aparatos que utilizamos tienen, hoy en día, acoplado un ordenador. Permiten obtener los datos de forma más rápida y cómoda, hacen aquellas operaciones que resultaban interminables y tediosas hace unos años; anotan los resultados, representan curvas, integran los picos, controlan la velocidad de medida, incluso otras variables, como temperatura, cantidad de CO_2 y de O_2 o intensidad de luz. Son verdaderamente útiles, y normalmente no se necesitan computadores de gran memoria para este tipo de operaciones.

MARGA.—Y nos olvidamos de su uso como elemento de organización del trabajo...

FEDE.—Sí, pero creo que por hoy ya es bastante. Recojamos nuestras cosas y vamos a tomar algo, creo que nos conviene olvidar los ordenadores un rato.

CARLOS.—Sí, creo que por hoy ya es bastante...

MARGA.—Y de la posibilidad de hacer archivos que ordenen las referencias, que controlen la llegada del material, que permitan hacer los protocolos...

FEDE y CARLOS.—¡Marga!

FE DE ERRATAS DEL TOMO

Pág.	Línea del programa	Donde dice	Debe decir
22	2050	PRINT cantidad;	PRINT #ca; cantidad;
74	10	GOSUB 8000	GOSUB 8770

ENCICLOPEDIA PRACTICA DE LA

INFORMATICA APLICADA

INDICE GENERAL

5 AÑADA PERIFERICOS A SU ORDENADOR

Breve descripción de varios periféricos que facilitan la comunicación con el ordenador personal, con algunos ejemplos de fácil construcción: ratón, lápiz óptico, marco para pantalla táctil...

6 GRAFICOS ANIMADOS CON EL ORDENADOR

En este libro las técnicas utilizadas para la animación son el resultado de unas pocas ideas básicas muy sencillas de comporender. Descubrirá los trucos y secretos de movimientos, choques, rebotes, explosiones, disparos, saltos, etc.

7 JUEGOS INTELIGENTES EN MICROORDENADORES

Los ordenadores pueden enfrentarse de forma «inteligente» ante puzzles y otros tipos de juegos. Esto es posible gracias al nuevo enfoque que ha dado la IA a la tradicional teoría de juegos.

8 PERIFERICOS INTERACTIVOS PARA SU ORDENADOR

Descripción detallada de la forma de construir, paso a paso y en su propia casa, dispositivos electrónicos que aumentarán la potencia y facilidad de uso de su ordenador: tableta digitalizadora, convertidores de señales analógicas, comunicaciones entre ordenadores.

9 COMO HACER DIBUJOS TRIDIMENSIONALES EN EL ORDENADOR PERSONAL

Compruebe que también con su ordenador personal puede llegar a diseñar y calcular imágenes en tres dimensiones con técnicas semejantes a las utilizadas por los profesionales del dibujo con equipos mucho más sofisticados.

10 PRACTIQUE MATEMATICAS Y ESTADISTICA CON EL ORDENADOR

En este libro se repasan los principales conceptos de las Matemáticas y la Estadística, desde un punto de vista eminentemente práctico y para su aplicación al ordenador personal. Se basan los diferentes textos en la presentación de pequeños programas (que usted podrá introducir en su ordenador personal).

11 CRIPTOGRAFIA: LA OCULTACION DE MENSAJES Y EL ORDENADOR

En este libro se presentan las técnicas de mensajes a través de la criptografía desde los primeros tiempos hasta la actualidad, en que el uso de los computadores ha proporcionado la herramienta necesaria para llegar al desarrollo de esta técnica.

12 APL: LENGUAJE PARA PROGRAMADORES DIFERENTES

APL es un lenguaje muy potente que proporciona gran simplicidad en el desarrollo de programas y al mismo tiempo permite programar sin necesidad de conocer todos los elementos del lenguaje. Por ello es ideal para quienes reúnan imaginación y escasa formación en Informática.

13 ECONOMIA DOMESTICA CON EL ORDENADOR PERSONAL

Breve introducción a la contabilidad de doble partida y su aplicación al hogar, con explicaciones de cómo utilizar el ordenador personal para facilitar los cálculos, mediante un programa especialmente diseñado para ello.

14 COMO SIMULAR CIRCUITOS ELECTRONICOS EN EL ORDENADOR

Introducción a los diferentes métodos que se pueden emplear para simular y analizar circuitos electrónicos, mediante la utilización de diferentes lenguajes.

15 COMO CONSTRUIR SU PROPIO ORDENADOR

Cuando se trabaja con un ordenador, lo único que puede apreciarse, a simple vista, es una especie de caja negra que, misteriosamente, acepta una serie de instrucciones. En realidad, un ordenador es una máquina capaz de recibir, transformar, almacenar y suministrar datos.

16 EL ORDENADOR COMO INSTRUMENTO MUSICAL Y DE COMPOSICION

Análisis de cómo se puede utilizar el ordenador para la composición o interpretación de música. Libro eminentemente práctico, con numerosos ejemplos (que usted podrá practicar en su ordenador casero) y lleno de sugerencias para disfrutar haciendo de su ordenador un verdadero instrumento musical.

17 SISTEMAS OPERATIVOS: EL SISTEMA NERVIOSO DEL ORDENADOR

Características de diversos sistemas operativos utilizados en los ordenadores personales y caseros. Se trata de llegar al conocimiento, ameno aunque riguroso, de la misión del sistema operativo de su ordenador, para que usted consiga sacar mayor rendimiento a su equipo.

18 UNIX, EL ESTANDAR DE LOS SISTEMAS OPERATIVOS MULTIUSUARIO

La aparición y posterior difusión del sistema operativo UNIX supuso una revolución en el mercado, de tal modo que se ha convertido en el estándar de los sistemas multiusuario. Su aparente complejidad podría provocar, en principio, un primer rechazo, pero debido a su potencia se convierte rápidamente en una extraordinaria herramienta de trabajo apta para cualquier tipo de aplicaciones.

19 EL ORDENADOR Y LA ASTRONOMIA

Los cálculos astronómicos y el conocimiento del firmamento en un libro apasionante y curioso.

20 VISION ARTIFICIAL. TRATAMIENTO DE IMAGENES POR ORDENADOR

El procesado de imágenes es un campo de reciente y rápido desarrollo con importantes aplicaciones en área tan diversas como la mejora de imágenes biomédicas, robóticas, teledetección y otras aplicaciones industriales y militares. Se presentan los principios básicos, los sistemas y las técnicas de procesado más usuales.

21 PRACTIQUE HISTORIA Y GEOGRAFIA CON SU ORDENADOR

Libro interesante para los aficionados a estas ciencias, a quienes presenta una nueva visión de cómo utilizar el microordenador en su estudio.

22 LA CREATIVIDAD EN EL ORDENADOR. EXPERIENCIAS EN LOGO

El LOGO es un lenguaje enormemente capacitado para la creación principalmente gráfica y en especial para los niños. En este sentido se han desarrollado numerosas experiencias. En el libro se analizan estas experiencias y las posibilidades del LOGO en este sentido, así como su aplicación a su ordenador casero para que usted mismo (o con sus hijos) pueda repetirlas.

23 EL LENGUAJE C, PROXIMO A LA MAQUINA

Lenguaje de programación que se está imponiendo en los microordenadores más grandes, tanto por su facilidad de aprendizaje y uso, como por su enorme potencia y su adecuación a la programación estructurada. Vinculado íntimamente al sistema operativo UNIX es uno de los lenguajes de más futuro entre los que se utilizan los micros personales.

24 BASIC

El lenguaje BASIC, es la forma más fácil de aprender las instrucciones más elementales con las que podemos mandar a nuestro ordenador que haga las más diversas tareas.

25 COMO ELEGIR UNA HOJA ELECTRONICA DE CALCULO

En este título se estudian las diferentes versiones existentes de esta aplicación típica, desde el punto de vista de su utilidad para, en función de las necesidades de cada usuario y del ordenador de que dispone, poder elegir aquélla que más se adecúe a cada paso.

26 BASIC AVANZADO

Una vez conocidas las instrucciones fundamentales del lenguaje BASIC, se plantea la cuestión de la realización de programas que resuelvan problemas o aplicaciones que se nos presentan diariamente en el trabajo, en casa o en los estudios.
Este libro trata de mostrar cómo se podrían realizar algunas de estas aplicaciones, estudiando diversas estructuras que proporciona el lenguaje BASIC (como las subrutinas) y viendo las ideas fundamentales para realizar gráficos en pantalla mediante un programa y para almacenar datos en discos o cintas mediante los ficheros.

27 APLIQUE SU ORDENADOR A LAS CIENCIAS NATURALES

Se abordan algunos temas interesantes de las Ciencias Naturales, desde un punto de vista eminentemente práctico y para su aplicación al ordenador personal. Se incluyen en el texto pequeños programas (que usted podrá introducir en su ordenador personal); estos programas se analizan y se indican posibles modificaciones a introducir para variar los resultados que se deben obtener. Libro útil tanto para que el estudiante de EGB o BUP repase estos conceptos que está aprendiendo, como para que el adulto recuerde algunos conocimientos olvidados que le interesan hoy.

28 PRACTIQUE FISICA Y QUIMICA CON SU ORDENADOR

Libro eminentemente práctico para realizar pequeños «experimentos» con su ordenador y distraerse de un modo útil.

29 EL ORDENADOR Y LA LITERATURA

En este libro se examinan procesadores de textos, programas de análisis literario y una curiosa aplicación desarrollada por el autor: APOLO, un programa que compone estructuras poéticas.

30 ERGONOMIA: COMUNICACION EFICIENTE HOMBRE-MAQUINA

Análisis de la comunicación entre el hombre y la máquina, y estudio de diferentes soluciones que tienden a facilitarla lo más posible.

31 LOS LENGUAJES DE LA INTELIGENCIA ARTIFICIAL

Libro en que se describen los lenguajes específicos para la «elaboración del saber» y los entornos de programación correspondientes. El conocimiento de estos lenguajes, además de interesante en sí mismo, es sumamente útil para entender todo lo que la Informática Artificial supondrá para el futuro de la Informática.

32 ¿MAQUINAS MAS EXPERTAS QUE LOS HOMBRES?

Después de situar los «sistemas expertos» en el contexto de la inteligencia artificial y describir su construcción, su funcionamiento, su utilidad, etc., se analiza el papel que pueden tener en el futuro (y presente, ya) de la Informática.

NOTA:
Ediciones Siglo Cultural, S. A., se reserva el derecho de modificar, sin previo aviso, el orden, título o contenido de cualquier volumen de esta colección.

Se abordan algunos temas interesantes de las Ciencias Naturales, desde un punto de vista eminentemente práctico y para su aplicación al ordenador personal. Se incluyen en el texto pequeños programas (que usted podrá introducir en su ordenador personal); estos programas se analizan y se indican posibles modificaciones a introducir para variar los resultados que se deben obtener. Libro útil tanto para que el estudiante de EGB o BUP repase estos conceptos que está aprendiendo, como para que el adulto recuerde algunos conocimientos olvidados que le interesan hoy.